AFRL-ML-WP-TR-2003-4116

PATHFINDER 2: IN SITU DESIGN COST TRADES (IDCT) TOOL

Allen G. Greenwood, Ph.D., P.E.

Mississippi State University (MSU)
260 McCain Engineering Building
P.O. Box 9542
Mississippi State, MS 39762

MAY 2003

Final Report for 21 July 2000 – 30 May 2002

Approved for public release; distribution is unlimited.

MATERIALS AND MANUFACTURING DIRECTORATE
AIR FORCE RESEARCH LABORATORY
AIR FORCE MATERIEL COMMAND
WRIGHT-PATTERSON AIR FORCE BASE, OH 45433-7750

NOTICE

USING GOVERNMENT DRAWINGS, SPECIFICATIONS, OR OTHER DATA INCLUDED IN THIS DOCUMENT FOR ANY PURPOSE OTHER THAN GOVERNMENT PROCUREMENT DOES NOT IN ANY WAY OBLIGATE THE U.S. GOVERNMENT. THE FACT THAT THE GOVERNMENT FORMULATED OR SUPPLIED THE DRAWINGS, SPECIFICATIONS, OR OTHER DATA DOES NOT LICENSE THE HOLDER OR ANY OTHER PERSON OR CORPORATION; OR CONVEY ANY RIGHTS OR PERMISSION TO MANUFACTURE, USE, OR SELL ANY PATENTED INVENTION THAT MAY RELATE TO THEM.

THIS REPORT HAS BEEN REVIEWED BY THE OFFICE OF PUBLIC AFFAIRS (ASC/PA) AND IS RELEASABLE TO THE NATIONAL TECHNICAL INFORMATION SERVICE (NTIS). AT NTIS, IT WILL BE AVAILABLE TO THE GENERAL PUBLIC, INCLUDING FOREIGN NATIONS.

THIS TECHNICAL REPORT HAS BEEN REVIEWED AND IS APPROVED FOR PUBLICATION.

Alan E Herner

ALAN HERNER
Project Engineer
Integration & Technology Branch
Manufacturing Technology Division

Charles E. Wagner

CHARLES WAGNER
Chief
Integration & Technology Branch
Manufacturing Technology Division

Copies of this report should not be returned unless return is required by security considerations, contractual obligations, or notice on a specific document.

REPORT DOCUMENTATION PAGE

Form Approved
OMB No. 0704-0188

1. REPORT DATE ***(DD-MM-YY)*** May 2003	**2. REPORT TYPE** Final	**3. DATES COVERED** ***(From - To)*** 07/21/2000 – 05/30/2002

4. TITLE AND SUBTITLE
PATHFINDER 2: IN SITU DESIGN COST TRADES (IDCT) TOOL

5a. CONTRACT NUMBER
F33615-00-C-5902

5b. GRANT NUMBER

5c. PROGRAM ELEMENT NUMBER
78011F

6. AUTHOR(S)
Allen G. Greenwood, Ph.D., P.E.

5d. PROJECT NUMBER
2865

5e. TASK NUMBER
01

5f. WORK UNIT NUMBER
50

7. PERFORMING ORGANIZATION NAME(S) AND ADDRESS(ES)

Mississippi State University (MSU)
260 McCain Engineering Building
P.O. Box 9542
Mississippi State, MS 39762

8. PERFORMING ORGANIZATION REPORT NUMBER

9. SPONSORING/MONITORING AGENCY NAME(S) AND ADDRESS(ES)
Materials and Manufacturing Directorate
Air Force Research Laboratory
Air Force Materiel Command
Wright-Patterson AFB, OH 45433-7750

10. SPONSORING/MONITORING AGENCY ACRONYM(S)
AFRL/MLMT

11. SPONSORING/MONITORING AGENCY REPORT NUMBER(S)
AFRL-ML-WP-TR-2003-4116

12. DISTRIBUTION/AVAILABILITY STATEMENT
Approved for public release; distribution is unlimited.

13. SUPPLEMENTARY NOTES
Report contains color.

14. ABSTRACT

The integration of cost evaluations into the design process, i.e., making them in situ, requires a strategic cost evaluation framework. This framework must address three key components: (1) the processes by which design and cost evaluations are performed, (2) the methodologies/technologies (M/T) that are needed in order to effectively carry out these processes, and more specifically, (3) the way cost models are used, reused, and managed. In order to design in affordability early in design, engineering tools, including cost assessment tools, must be integrated into the design process and interact with tools that support other types of, but related, analyses. Insertion of these technologies requires an understanding of the design, requirements, and cost-estimation processes. Therefore, definition of these key processes, and the relationships among the processes, are essential for defining and developing an integrated support tool to enhance these processes. There are many M/T available and being developed to support cost evaluations during design; however, they are not a part of an overall strategy or plan that considers the needs and requirements of the design trade study process and the relationships among the M/T. Models are an important element in cost estimating; they transform parameters and design variables into estimates of cost performance and risk.

15. SUBJECT TERMS
design

16. SECURITY CLASSIFICATION OF:			**17. LIMITATION OF ABSTRACT:**	**18. NUMBER OF PAGES**	**19a. NAME OF RESPONSIBLE PERSON** (Monitor)
a. REPORT	**b. ABSTRACT**	**c. THIS PAGE**	SAR	258	Alan E. Herner
Unclassified	Unclassified	Unclassified			**19b. TELEPHONE NUMBER** *(Include Area Code)* (937) 904-4399

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

EXECUTIVE SUMMARY

The integration of cost evaluations into the design process, i.e. making them in situ, requires a strategic cost evaluation framework. This framework must address three key components: (1) the processes by which design and cost evaluations are performed, (2) the methodologies/technologies (M/T) that are needed in order to effectively carry out these processes, and more specifically (3) the way cost models are used, reused, and managed. In order to design-in affordability early in design, engineering tools, including cost assessment tools, must be integrated into the design process and interact with tools that support other types of, but related, analyses. Insertion of these technologies requires an understanding of the design, requirements, and cost-estimation processes. Therefore, definition of these key processes, and the relationships among the processes, are essential for defining and developing an integrated support tool to enhance these processes. There are many good methodologies/technologies (M/T) available and being developed to support cost evaluations during design; however, they are not a part of an overall strategy or plan that considers the needs and requirements of the design trade study process and the relationships among the M/T. Models are an important element in cost estimating; they transform parameters and design variables into estimates of cost performance and risk. Given their importance, there are many key issues related to cost estimating models.

In order to help address these needs this project had three primary initiatives:

1) begin developing the foundation for a strategic “view” or framework of the design/cost processes and the needs that exist for cost evaluations to effectively support design tradeoff studies by initiating the:
    a) definition of the design, requirements engineering/management, and cost assessment processes and develop a means to represent these processes,
    b) identification and definition of the needs that exist for effectively carrying out the design/cost processes; in addition, develop a means to structure or represent these needs,
    c) identification and definition of the capabilities that exist or that are evolving that are and can be applied to address the need to effectively carry out the design/cost processes; also develop a means to represent these capabilities, and
    d) mapping of needs and capabilities in order to identify “gaps,” as well as alternative means.
2) utilize the process definitions and needs assessment to develop an architecture for integrating the M/T to better support design tradeoff studies.
3) develop prototype applications of the IDCT Tool through iterative design and industry reviews in order to provide proof of concept and demonstrate components of the architecture. The Cost Risk Tool prototype, version 3, has the following capabilities:
    a) based on design and cost-analyses processes
    b) object-oriented, model centric architecture (e.g. model registration and management, variable complexity models)
    c) flexible, coupled, dynamic, hierarchical product- and cost-breakdown structures
    d) integrated Monte Carlo simulation engine to facilitate risk analysis
    e) simulation-based reliability prediction model.

The project was progressing in all three initiatives, as is demonstrated in the contents of this report, when the project was curtailed. Dissemination of results to date is through at least 3 journal articles (1 published, 2 in preparation) and 3 conference papers.

This report is organized into four sections:

- Section 1: Project Description. An overview of the problem addressed and project objective and goals, as well a summary of project activities, participants, and key results. More details on many

of the key activities are presented in the following section in the form of papers and technical notes.

- Section 2: Conference/Journal Articles & Technical Notes. Discussion of many of the topics presented in Section 1 are expanded in this section through papers that have been written about work that has been performed as part of this project. Some of the papers are published conference or journal articles; others are submitted or nearly ready for submission. This section also includes technical notes on topics that were researched in this project; the information is provided in note format because some are too short to be published, while others are not yet in an appropriate form for publication.
- Section 3: Project Briefing. Presentation slides that summarize the work performed in the project.
- Section 4: Foundation Report. Since the precursor study to this project was not published by the Air Force, and since the information in the study provides the basis for the work being reported, the precursor report is included as the last section of this report. The precursor report was completed by the same Principal Investigator in September 2000, for AFRL through a contract with Anteon Corporation, Contract F33615-96-D-5608, Delivery Order No. 34.

1. PROJECT DESCRIPTION

This project was funded by the Air Force Research Laboratory (AFRL) from July 2000 through May 2002; it was co-funded by Mississippi State University (MSU) from July 2000 to the present.

This report is organized into four sections:

- Section 1: Project Description. An overview of the problem addressed and project objective and goals, as well a summary of project activities, participants, and key results. More details on many of the key activities are presented in the following section in the form of papers and technical notes.
- Section 2: Conference/Journal Articles & Technical Notes. Discussions of many of the topics presented in Section 1 are expanded in this section through papers that have been written about work that has been performed as part of this project. Some of the papers are published conference or journal articles; others are submitted or nearly ready for submission. This section also includes technical notes on topics that were researched in this project; the information is provided in note format because some are too short to be published, while others are not yet in an appropriate form for publication.
- Section 3: Project Briefing. Presentation slides that summarize the work performed in the project.
- Section 4: Foundation Report. Since the precursor study to this project was not published by the Air Force, and since the information in the study provides the basis for the work being reported, the precursor report is included as the last section of this report. The precursor report was completed by the same Principal Investigator in September 2000, for AFRL through a contract with Anteon Corporation, Contract F33615-96-D-5608, Delivery Order No. 34.

Section 1 is organized as follows. The first subsection describes the basic problem that this project addresses. This is followed by a statement of the project's objective and goals, a discussion of the underlying philosophy that provides the foundation for the concepts and methodologies that have been developed, and a brief description of previous work. This is followed by a summary of activities carried out during this project, a description of the cost-risk tool prototype that was developed, a list of project participants, and a summary of accomplishments.

The reference section at the end of Section 1 contains only the sources specifically referenced in this section. Most of these refer to the publications included in Section 2. These publications provide the details of the concepts and methodologies that were developed in this project and contain their own set of references. Section 4, the Foundation Document, also has its own set of references, which is much more comprehensive since that study involved a literature review of the field.

A list of acronyms is provided as Appendix 1-A.

1.1 Statement of the Problem

The design of affordable products requires cost evaluations to be an integral part of design tradeoff studies The integration of cost evaluations into the design process, i.e. making them in situ, requires a strategic cost evaluation framework. This framework must address three key components: (1) the processes by which design and cost evaluations are performed, (2) the methodologies/technologies (M/T) that are needed in order to effectively carry out these processes, and more specifically (3) the way cost models are used, reused, and managed.

In order to design-in affordability early in design, engineering tools, including cost assessment tools, must be integrated into the design process and interact with tools that support other types of, but related, analyses. Insertion of these technologies requires an understanding of the design, requirements, and cost-estimation processes. Therefore, definition of these key processes, and the relationships among the processes, are essential for defining and developing an integrated support tool to enhance these processes. The following identify some of the primary process-definition issues.

- There is no overall systems view of cost evaluation and related processes.
- There is a lack of a life-cycle perspective; i.e., there is little definition as to how existing M/T are used over the design and product life cycles and how the M/T are transitioned over the life cycle
- The main processes that use and provide information to the cost evaluation process are not well understood. As a result, they are not well defined. A larger concern is that the cost estimating / evaluating process itself is not well defined.
- Since the processes are not well defined, the relationships among the processes are not well defined.
- Also, since the processes are not well understood and defined, the needs of the processes are not clearly articulated.
- Without a clear definition of needs, capabilities, and processes, the varied technologies employed to support cost evaluations are not as effective as they could be. This lack of structure and strategy also contributes to a rather ad hoc technology development environment.
- Tools that support the cost evaluation process and the design process are often developed without a clear idea of how they fit into the overall process.

There are many good methodologies/technologies (M/T) available and being developed to support cost evaluations during design; however, they are not a part of an overall strategy or plan that considers the needs and requirements of the design trade study process and the relationships among the M/T. As a result, with regard to current and evolving M/T, there is:

- little understanding of how the M/T are related to one another and how they complement each other; basically there are islands of M/T for performing cost assessments. Oftentimes, M/T tend to address some local problem for one activity in the design tradeoff process and there is no clear definition of how they interface with other tools.
- no defined set of capabilities that are needed for cost assessments in order to effectively support the design trade study process, especially during early design.
- no formal “inventory” of capabilities that are addressed by existing and evolving M/T.
- no formal assessment and identification of the key “gaps” between M/T capabilities design/cost process needs; this gap analysis should provide the basis for effective resource allocations to address the gaps.
- M/T are disparate; there is no unifying structure or framework and no integration strategy for federating multiple M/T.

Models are an important element in cost estimating; they transform parameters and design variables into estimates of cost performance and risk. Given their importance, there are some key issues related to cost estimating models; generally, models:

- are naturally disparate – they address different needs at different stages of the development process. Unfortunately, there is no unifying structure or framework and no integration strategy for federating multiple models, regardless of their type.
- are not well linked to one another. Whether for the same estimating instance or over time, and from one analysis to another, they are isolated and result in islands of analysis.
- and data often become embedded in the analysis, rather than being external and available for multiple analyses.
- lack a general organizational structure, i.e. they are not organized in a manner like data in a database.
- data, and technologies tend to be focused on point solutions; i.e. they are used to address near-term and local questions or tradeoff studies and are not candidates for reuse on comparable projects.
- do not adequately address cost risk in design tradeoff studies.
- are not applied in a manner that leverages all of the information that is available about a system. Too often the cost structure and type of estimating model are applied to all components in the system rather than being based on the information that is available to perform the estimate; this is mostly due to the lack of a model-focused framework that facilitates the application of variable-complexity models. For example, a new technology may be introduced into part of the system and since little is known about the technology the use of a parametric cost model is appropriate. On the other hand, another part of the system may be nearly off-the-shelf, in which case a detailed buildup type of estimate is the most appropriate. The level of fidelity in the estimate is much greater for the latter system component. In this case, one should be able to select the most appropriate model based on the information that is available and link models of different types to various system components. Similarly, one should be able to change the models linked to the system as the design evolves and the information matures.

While the above list of issues is quite extensive, and beyond the scope of any one project, it is hoped that this project provides a small step towards confronting, addressing, and resolving some of these issues.

1.2 Project Objective and Goals

The overall objective of this effort is enhance the design of affordable products by providing, as an integral part of the trade-study processes, timely and relevant evaluations of the cost to produce and operate a product. It also is intended to foster a better understanding of the impacts of design and programmatic variables on product/process costs. This project is intended to address the problems outlined in the previous section. Specifically, this project was intended to:

- develop a framework, set of processes, and ultimately a design decision-support tool that would be an open and flexible system for conducting and managing cost analyses and trade studies early in the product design process,
- assimilate and integrate distributed and disparate cost-evaluation processes, data, and models, as well as advanced design technologies, for use over the product life cycle,
- develop a means to utilize the "most appropriate" models and data to assess product alternatives as the design evolves,
- identify and provide means for effectively inserting the technologies into the design process, and
- test, evaluate, and deploy the results of the project in industry and academe.

1.3. Underlying Philosophy

Prior to discussing the specific activities and outcomes of this project, we provide a brief introduction of the underlying philosophy that is the basis for this work.

This project is based upon the general notions that cost analyses:

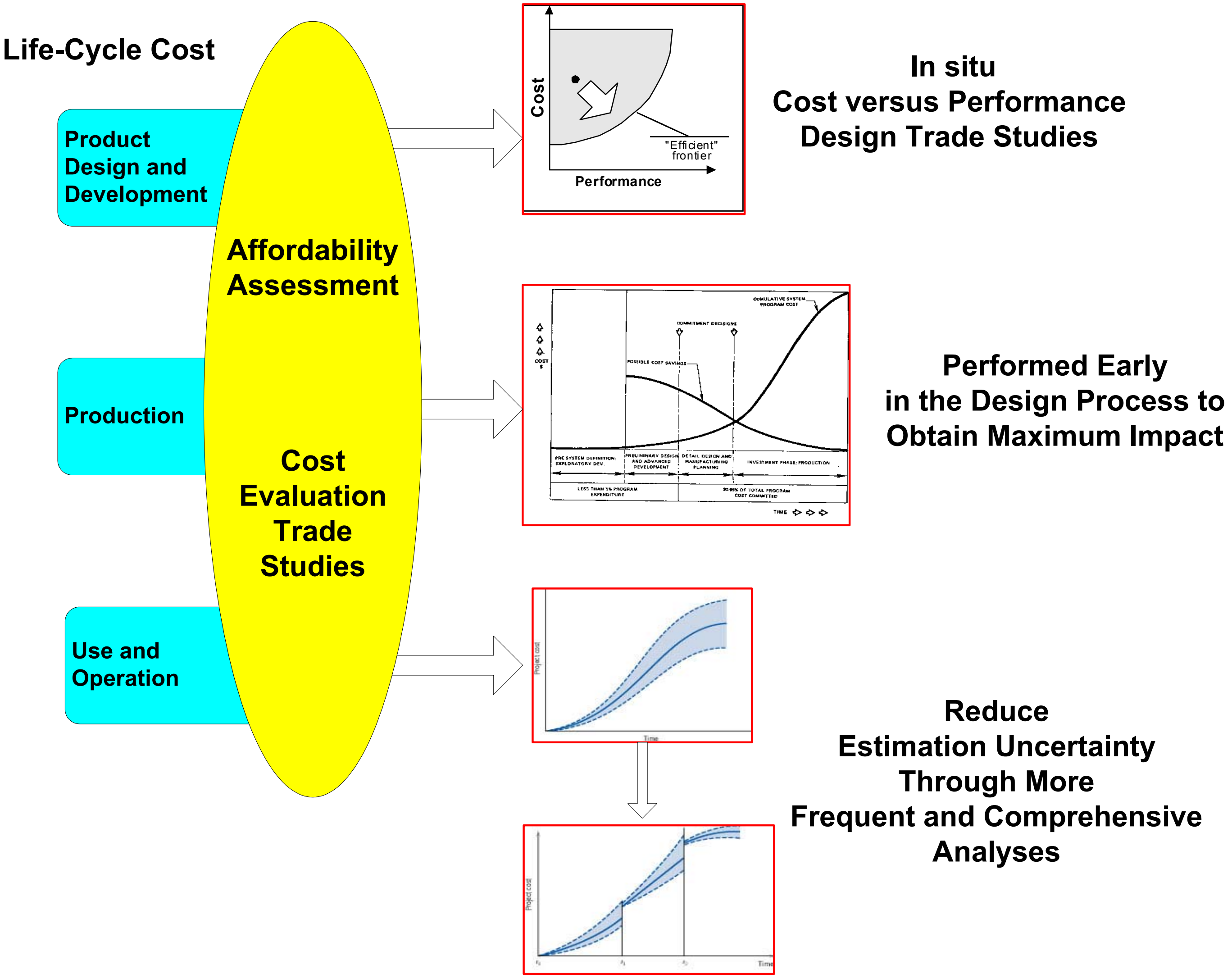
1. are a critical aspect of designing affordable products,
2. are a key link in the trade-off study process, providing feedback on the implications of design decisions,
3. need to be performed early in the design process to be most effective, i.e., to have the most impact,
4. link and transform design variables and parameters to measures of system performance,
5. are model based, i.e. models are the means to transform design variables and parameters into performance measures,
6. are performed according to definable processes and these processes are related to other engineering processes, such as design, requirements definition, etc.
7. need to address life-cycle cost (research and development, production, operations and support) implications,
8. need to be linked with previous analyses for the system being developed as well as with related systems in order to facilitate reuse, leverage related knowledge, track change, etc.,
9. need to include assessments of risk and uncertainty,

The next several figures are used to illustrate these concepts. First, Figure 1-1 shows that the IDCT Tool is aimed at fulfilling a critical need in the design of affordable products. It is used to facilitate and enhance design decisions by trading off performance and life-cycle cost (LCC) – including product design and development, production, and operations and support -- early in the design process. It not only considers “point estimates” for the cost of design alternatives but the cost risk and uncertainty of the estimate.

In order to design and develop affordable products, design and cost activities need to be tightly linked. This is represented in the upper left portion of Figure 1-2. Both activities are driven by requirements and capabilities. At the highest level, design activities provide both production and operations characteristics/attributes of the design to the cost analysis activities; in return the cost analyses provide the design team with measures of cost performance.

Our approach calls for cost analyses to be a variable-complexity design decision-support system (DSS). This is illustrated in the lower portion of Figure 1-2 by the three ellipses that represent the three primary components of a DSS – models, data, and user interfaces. Enabling technologies are noted as well since they are used within and among the DSS components and are a required to implement the DSS. We posit that the design/cost decision-support technologies must:

- consider uncertainty
- be able to utilize a wide variety of models that depend on the information available
- be able to utilize data from multiple sources, multiple systems, and from various parts of the life cycle, including historical design and cost studies.



Life-Cycle Cost

Product Design and Development

Production

Use and Operation

Affordability Assessment

Cost Evaluation Trade Studies

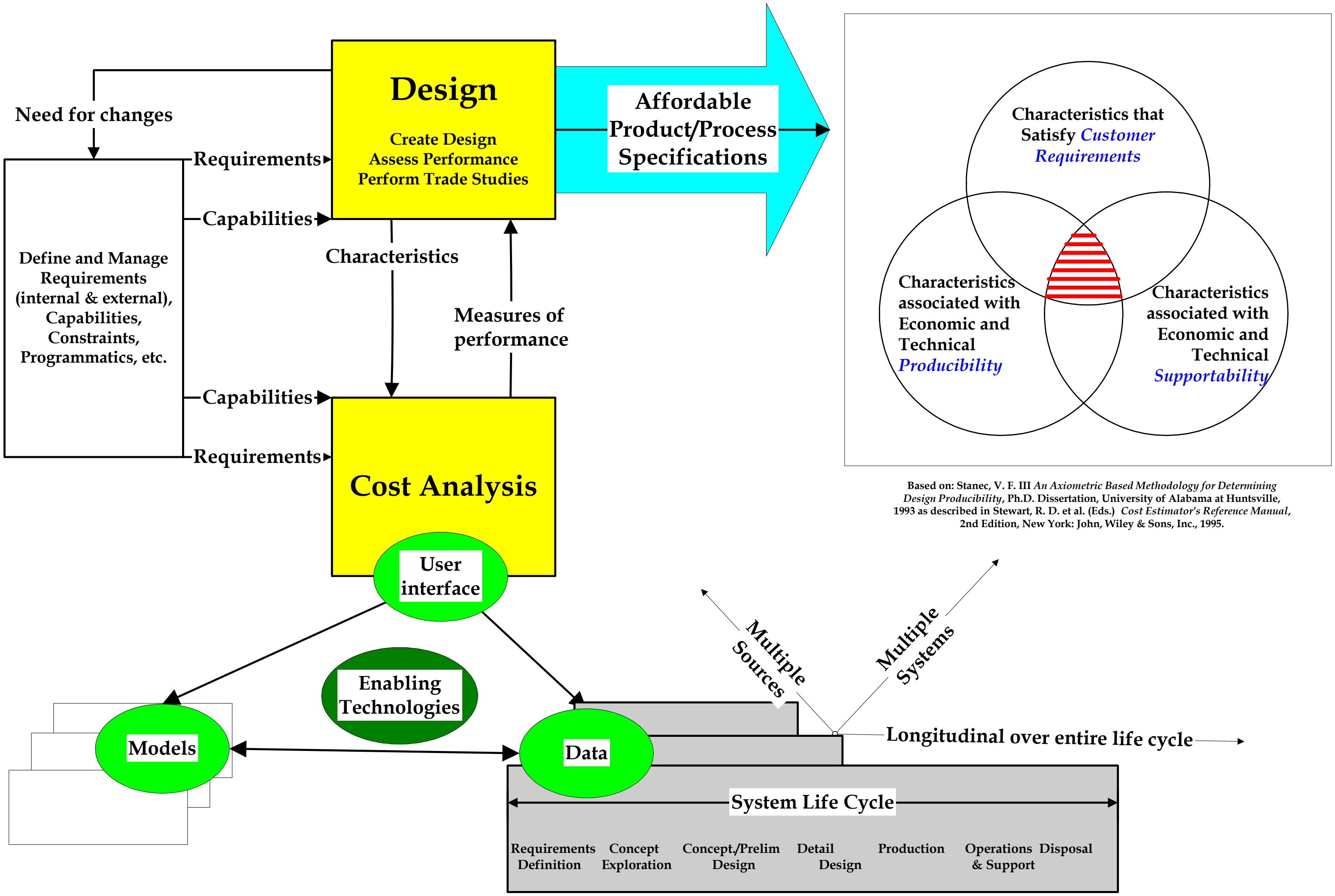
In situ
Cost versus Performance
Design Trade Studies

Performed Early
in the Design Process to
Obtain Maximum Impact

Reduce
Estimation Uncertainty
Through More
Frequent and Comprehensive
Analyses

Cost
"Efficient" frontier
Performance
COST $
CUMULATIVE SYSTEM PROGRAM COST
COMMITMENT DECISIONS
POSSIBLE COST SAVINGS
PRE SYSTEM DEFINITION: EXPLORATORY DEV.
PRELIMINARY DESIGN AND ADVANCED DEVELOPMENT
DETAIL DESIGN AND MANUFACTURING PLANNING
INVESTMENT PHASE: PRODUCTION
LESS THAN 5% PROGRAM EXPENDITURE
90-95% OF TOTAL PROGRAM COST COMMITTED
TIME
Project cost
Time
Project cost
Time

Figure 1-1: IDCT Tool meets a critical need to trade off performance, cost, and risk early in the design process.



Based on: Stanec, V. F. III *An Axiometric Based Methodology for Determining Design Producibility*, Ph.D. Dissertation, University of Alabama at Huntsville, 1993 as described in Stewart, R. D. et al. (Eds.) *Cost Estimator's Reference Manual*, 2nd Edition, New York: John, Wiley & Sons, Inc., 1995.

Figure 1-2: Design performance should be assessed through a decision-oriented LCC tool.

In Figure 1-2, the data component of the DSS overlaps a timeline that represents a system's life cycle. This illustrates that cost analyses are performed longitudinally throughout a product's life cycle; although it is well known that in order to have the most impact, LCC should be addressed early in the design process. The figure also represents the notion that cost analyses should utilize data from multiple sources and multiple systems at varying stages in the life cycle. This supports the notion of variable complexity in that the analyses reflect the fidelity of the design; that is, each component within a system should be analyzed based on the level of definition of that component. For example, components of a system in conceptual design may be mature and fielded; therefore, detailed, highly reliable cost data should be available and as a result should be used. However, other components within in the same system may utilize new technologies; therefore, cost data most likely is not available and must be estimated with high-level models, e.g., using parametric methods. Therefore, the DSS should have access to a variety of models and data and the appropriate models and data should be invoked depending on the type and extent of information that is available. The models and data that are used should change as the design evolves, reflecting the increasing definition of the design.

At the top of Figure 1-2, the arrow that emanates from the design activity leads to three intersecting circles that represent customer requirements, producibility, and supportability. The shaded area at the intersection of the circles denotes that a successful design will lie within the intersection where customer requirements are satisfied, the product is economically and technically producible, and is economically and technically supportable. The aggregation of these characteristics is often reflected in life-cycle cost.

Prior to developing a DSS for assessing the cost of alternative designs, it is important to understand that cost analysis is a process and clearly define the activities associated with that process. Figure 1-3 represents the cost analyses process in an IDEF$_0$ format; i.e., as is described further below, the Cost Analysis box in the center of the figure is represented as an ICOM box where input, controls, output, and mechanisms are denoted on each of the four sides of the box.

The contents of the Cost Analysis box in Figure 1-3 identify the key types of analyses that are typically performed: estimating (providing performance measure(s) for a single design), comparing (simultaneous consideration of multiple designs with a focus on similarities), contrasting (simultaneous consideration of multiple designs with a focus on differences), sensitivity analyses, and optimization. The tool developed in this project focuses on estimating and sensitivity analyses but provides the foundation for extending its capability to include the concurrent assessment of multiple designs.

Cost analyses are fundamentally transformation processes, converting a variety of characteristics into performance measures. The inputs to the cost analysis process are shown along the left side of the ICOM box in Figure 1-3 and include programmatics (program delivery schedule, purchase quantity, etc.), cost factors/parameters (e.g. labor rates, escalation factors), physical characteristics (e.g., size, weight, material, reliability), production characteristics (e.g. primary production processes, scrap rate, learning curve factors), and operations characteristics (e.g. mission profile, maintenance plan). All of these are received in some form from domain experts. Controls are denoted at the top of the ICOM box and include the phase of the life cycle when the analysis is performed, and the resolution or fidelity of the design. Output from cost analyses are primarily evaluation measures such as specific cost values, as well as measures of cost risk. They are conveyed to the design team and used assess the impact of design changes. Mechanisms are things used to carry out the process activities; they include data, models and tools, and domain experts.

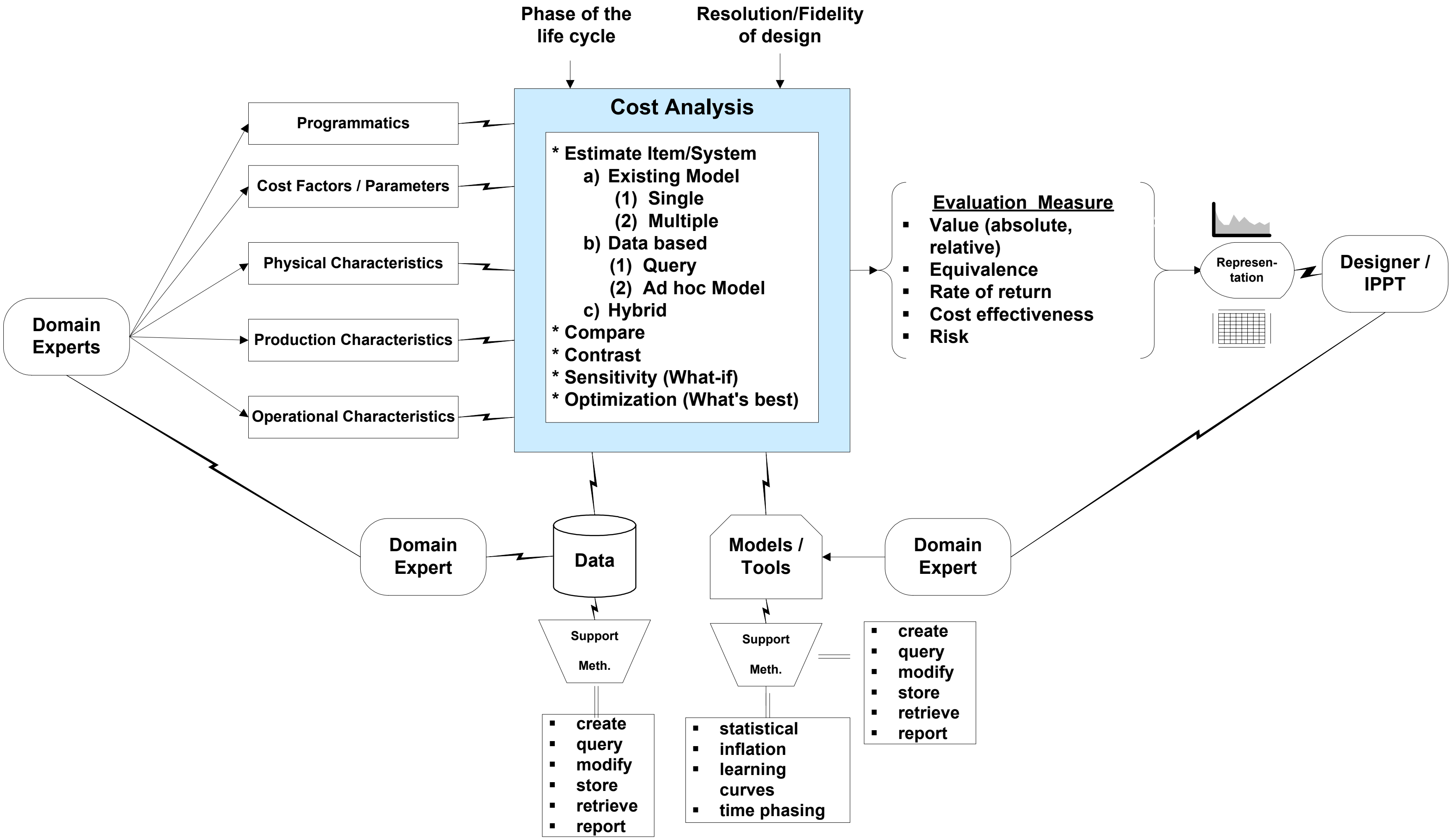


Figure 1-3: A systems view of cost analyses.

Understanding and defining cost analysis activities is a necessary but not sufficient condition for the design and development of an effective design/cost DSS. The design activities must be defined as well and the relationships among the design and cost activities must be understood and defined. As a start towards that end, the relationship map in Figure 1-4 defines some of the high-level activities that are typically undertaken during a cost analysis and identifies some preliminary links to the design process. The activities in Figure 1-4 are grouped into five primary activities – define, set up, decide, determine, and trade off. The key activity is "trade off," where the cost and performance of alternative designs are traded off in order to determine the "best" approach. Obviously, cost is a major part of the trade-off study.

All of the other primary activities support the trade-off activity. The design process provides an item structure, a hierarchical representation of the product, and a set of characteristics; these, along with programmatic information from the Define activity, are transformed, typically through the use of some type(s) of models, into cost measures. The transformation takes place within the Determine activity.

The Setup and Decide activities shown in Figure 1-4 are used to establish the costing strategy that is applied to the analysis of a particular design. The Setup activity identifies the types of cost that will be considered -- e.g. manufacturing, fly-away, life-cycle-- the cost elements, and their structure. It also identifies the cost models that are available and associates each model with a cost element or set of cost elements. The Decide activity defines both the costing approach and the costing methodology and is performed for each trade-off study; i.e., different components in a design, and even different alternatives for those components, may require a different costing approach or methodology. For example, a fielded component involves little uncertainty and there should be sufficient information for a detailed buildup of costs; therefore, the approach would be deterministic and the methodology would be a detailed engineering build up. This is in contrast to a new technology, where the approach would be stochastic to assess the inherent risk and the methodology would likely be parametric.

1.4 Previous Work

This project builds upon previous work performed by the Principal Investigator. It most closely builds upon a precursor or foundation study that directly preceded this project. Since the report from the foundation study was never published, it is included as Section 4. A brief summary of the precursor project is provided below, along with its conclusions.

Since this is the initial step towards the development of a comprehensive design decision-support tool, a significant portion of the project focused on defining user needs and system requirements. As a prerequisite to these activities, the product life cycle and systems development processes were defined and the general characteristics of cost-evaluation environments were identified. All of these activities provided the foundation for identifying user needs and establishing system requirements. In addition, a preliminary concept of operation for the Tool was formulated.

A conceptual design for an IDCT Tool was developed using an object-based approach. The design scheme focused on the intelligent integration of cost elements (models, data, tools, etc.) in order to provide an effective decision support system for cost evaluation during design, especially during the conceptual and preliminary design phases. Based on the conceptual design for the IDCT Tool, commercial off-the-shelf (COTS) software were identified that could support the development of the Tool. In order to demonstrate feasibility of the approach and design, we developed a prototype of an IDCT Tool.

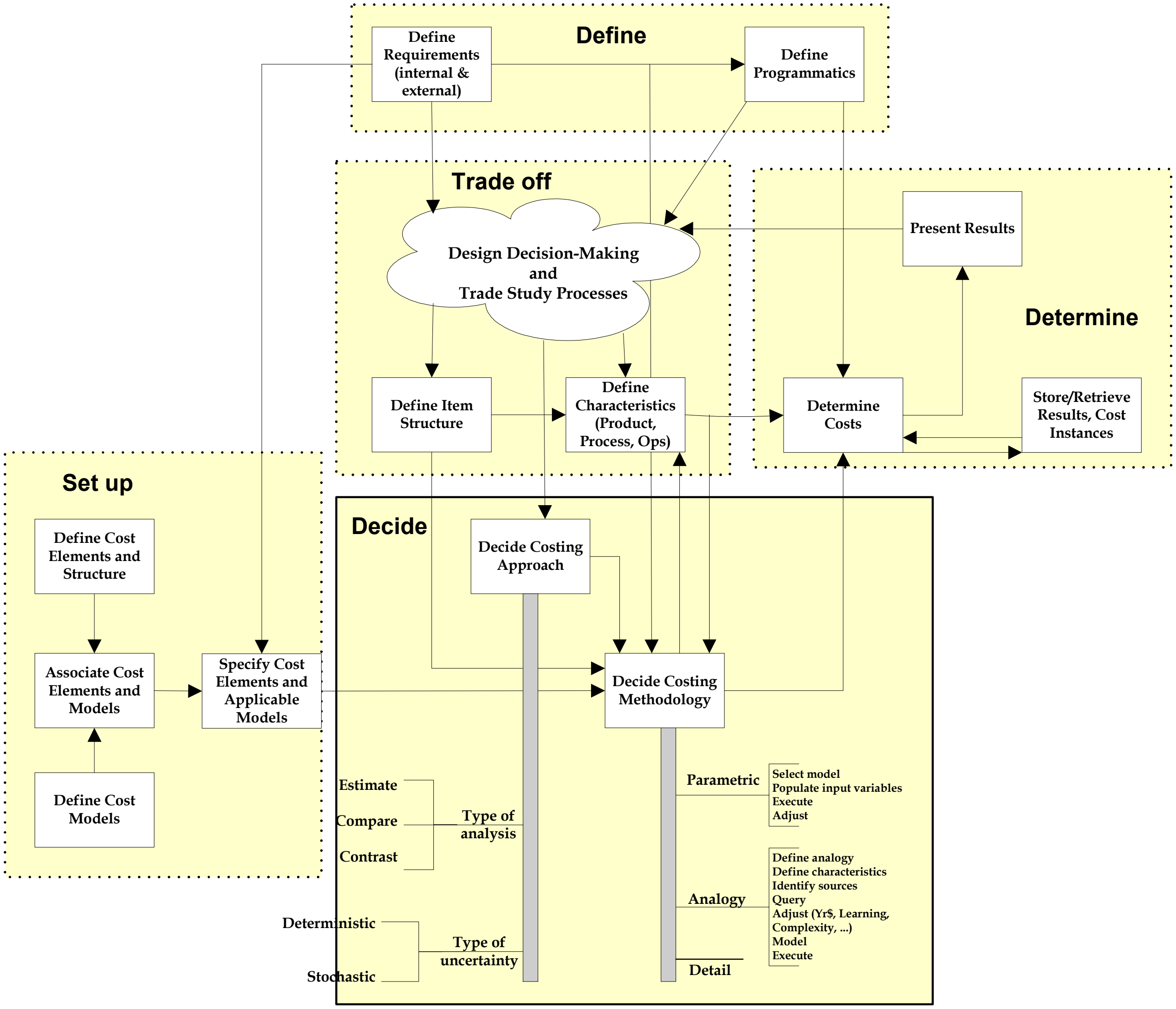


Figure 1-4: Cost analysis activities relationship map.

All of the activities of the project provide a solid foundation for further development of the IDCT Tool, including a master plan for further development of the Tool and potential team members that would be a part of the development.

The following are the conclusions from the precursor study.

1. The cost-evaluation environment, especially during conceptual and preliminary design is often not well understood, disparate and highly cross functional, involves information from all phases of the product life cycle, is highly time sensitive (assessments need to be made as the design evolves, not after the fact), involves technologically non-homogeneous designs, is dynamic and uncertain, knowledge based, model centric, is highly concerned with data relevancy and currency, and involves many islands of technologies that need to be assimilated and synthesized.
2. The above characteristics of the design/cost-evaluation environment provide major challenges to the development of a comprehensive design decision support system. It also provides great opportunities to improve current practices and processes.
3. Despite the difficult problem, this project demonstrates the feasibility of defining, developing, and deploying an effective system that can provide cost evaluation support to the design trade study processes.
4. The critical foundation for the successful development and use of an IDCT system is a thorough understanding and definition of the design decision-making processes that involve cost evaluation, definition of the cost-evaluation processes that support design, and the relationships between the design and cost-evaluation processes.
   - It is through these processes that the needs of the stakeholders are identified.
   - The greatest opportunity for improvement is often at the interfaces between processes, i.e., through better communications, control, and management.
   - Design decisions should be based on information that is drawn from all applicable sources and programs and from throughout their life cycle; the models that are used in the decision-making processes should be based on the information that is available.
5. Real value can be added by the development of an IDCT Tool if, in addition to opening the design space, it provides a means to identify and understand *how* and *why* design variables influence cost. Similarly, an effective IDCT Tool should capture and utilize evolving design experiences as a means to learn from the experiences.
6. In order to produce affordable products, cost analyses need to be an integral part of conceptual design. To this end, cost analyses need to directly support and facilitate design decision-making (trade studies) processes. In order for this to effectively occur, cost analyses need to be:
   - timely (opportune) – provide feedback as the design evolves, not after the fact.
   - inclusive (comprehensive) – provide, as appropriate, measures of the product's cost that consider production and operations costs, direct and indirect costs, and recurring and non-recurring costs.
   - useful (relevant) – be sensitive to the needs of the users; e.g., provide feedback on the effect of programmatic variables on cost to the program manager and affect of design variables on cost to the designer.
   - representative (accurate) – based on the most applicable data, models, knowledge, etc.
   - non-intrusive -- be integrated into work processes, provide effective user interfaces, facilitate maintenance.
7. The fundamental capabilities that are needed by an IDCT Tool in order to effectively support early design decisions include:
   - assimilating existing and emerging cost evaluation and supporting technologies (costing systems, models, databases, design guidance, etc.) and providing an open system for easily incorporating new technologies,
   - managing cost-evaluation information over the product life cycle,

- utilizing the most recent and most appropriate data, models, etc. that are available and basing the choice of data, models, etc. on the type of design/programmatic information that is currently available,
- capturing evolving design experiences for the purpose of reuse and learning, and
- providing design guidance from a cost perspective, both on the cost to produce the product and the cost to use the product.

1.5 Activities

The following is a summary of activities that were performed during the project. Results from many of these activities are described in detail in the Conference/Journal Article & Technical Notes section and in the Briefing section of this report. Discussion of the activities is divided into the following sections: tool design, tool development, process definition, investigation of supporting technologies, and dissemination of results.

1.5.1 Tool Design

Tool design includes assessment of the prototype from the precursor study, definition of the IDCT Tool as a decision support system, and definition of the Tool's overall architecture.

1.5.1.1 Assessment of the prototype from the precursor study.

The work directly related to the prototype developed in the foundation study was reviewed and assessed in order to define the most effective way to proceed with the development of subsequent prototypes that are a part of this project. As a result, we decided to:

1) focus on developing a tool that would have broader and more general applicability, that is:
   a) support life cycle analyses rather than only manufacturing
   b) handle variable complexity models, i.e. be able to apply multiple models within an analysis based on the fidelity of the information available
   c) provide risk assessment capabilities, e.g. means to identify high cost and high cost risk components
   d) provide flexibility at the component level in terms of the types of costs considered and the level of detail
   e) institute structure and rigor into the cost evaluation process in order to permit the use of multiple "external" models, enhance tracking and documentation among alternatives, and encourage reuse in order to expand the design space. In essence, separate models and data from the analysis structure
   f) be MS Windows based
2) investigate means to manage the tool development process.

1.5.1.2 Definition of the IDCT Tool as a decision support system (DSS)

As described earlier, in order to be a successful tool for enhancing the design of affordable products, the IDCT Tool must leverage enabling technologies and integrate models, data, and users. As shown in Figure 1-5, the IDCT Tool is the core that leverages enabling technologies in order to effectively link users, models, and data. The "bubbles" emanating from the components of a DSS in Figure 1-5 provide a sampling of the issues that are associated with developing an IDCT DSS Tool. The "dashed bands" that wrap the four components represent the different types of analyses that the system supports and are referred to as use cases.

Basic DSS concepts were researched in this project. However, since models are a major focus of the project and model management is considered a major aspect of the IDCT Tool, these topics were investigated more heavily than the other DSS components.

The PI attended the INFORMS "Optimizing the Extended Enterprise in the New Economy" Conference in San Diego in May 2001. The aspects of the conference that were most closely related to this project were presentations on the recent developments in decision analysis and an introduction to the AIMMS (Advanced Integrated Multidimensional Modeling System) software from Paragon Decision Technologies. AIMMS provides a development environment for interactive decision support systems that use advanced computational techniques.

1.5.1.3 Architecture

A concept for a new underlying structure for the IDCT Tool was formulated. The model/data management structure is composed of an array for each design alternative; each cell in the array links component data to each element of the cost structure and the Tool's model base. This is an important element for effectively managing cost estimates and models. Some of the key aspects of the architecture include:

- concept of an "estimate instance" that was defined to merge two dynamic hierarchies – product breakdown structure and cost breakdown structure – and directly link to a model base.
- "model/estimate management structure" was developed to implement to the estimate instance concept.
- model-focused philosophy was implemented through the development of means for model selection, model registration based on a data dictionary, flexible cost structures, etc.
- use cases definitions to support the development of the IDCT Tool.
- object-based system architecture; represented in terms of an entity-relationship diagram.
- Monte Carlo simulation engine developed and incorporated in order to generate data for cost risk analysis. The consideration of risk and uncertainty of cost estimates is critical component in performing tradeoff studies early in the design process. A considerable amount of research went into the development of CRT's Monte Carlo simulation engine that estimates system risk based on component design parameters.

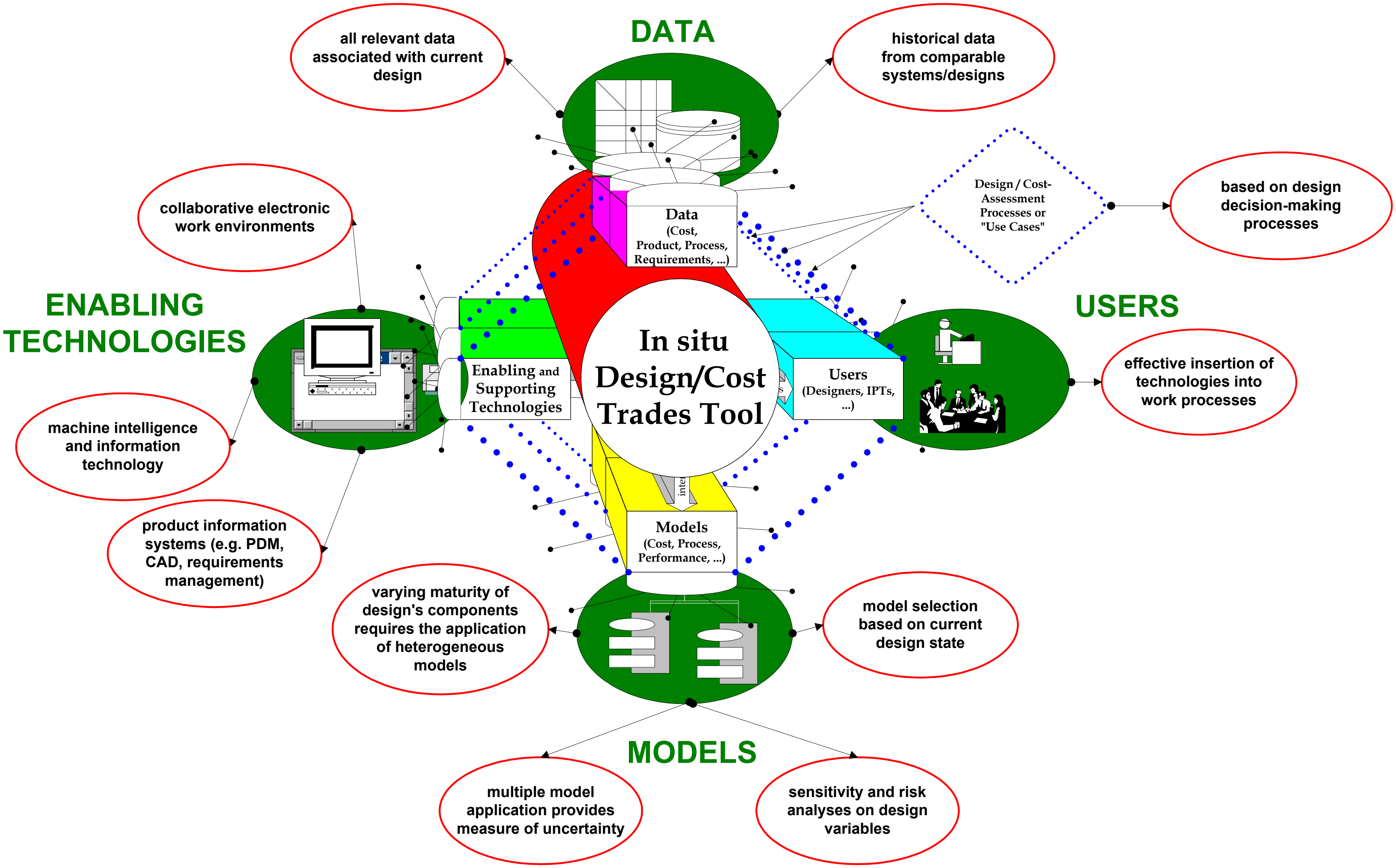

DATA
all relevant data associated with current design
historical data from comparable systems/designs
Design / Cost-Assessment Processes or "Use Cases"
based on design decision-making processes
collaborative electronic work environments
ENABLING TECHNOLOGIES
Data (Cost, Product, Process, Requirements, ...)
Enabling and Supporting Technologies
In situ Design/Cost Trades Tool
Users (Designers, IPTs, ...)
USERS
effective insertion of technologies into work processes
machine intelligence and information technology
product information systems (e.g. PDM, CAD, requirements management)
Models (Cost, Process, Performance, ...)
inte
varying maturity of design's components requires the application of heterogeneous models
model selection based on current design state
MODELS
multiple model application provides measure of uncertainty
sensitivity and risk analyses on design variables

Figure 1-5: Decision support approach and example issues.

In order to understand the current state of the art in distributed architectures, the PI attended the Enterprise Architecture Conference in New Orleans in March 2002. Several methodologies/technologies that are applicable to the IDCT Tool were identified, including information architecture frameworks, information management, and web portals. Funding for attendance at this conference was provided through another project.

The architecture is further explained in Section 1.6, Cost Risk Tool, and in the paper by Greenwood and Ormon, entitled “A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk.” It was presented at the Decision Sciences Institute Conference in November 2002 in San Diego and published in the conference’s proceedings; the full paper is included in Section 2.2.

1.5.2 Tool Development

Areas addressed within the tool development activities include the development process, the development environment, the Cost Risk Tool’s (CRT) releases and functionality, and an industry survey to provide a preliminary assessment of the project and Tool prototype.

1.5.2.1 Development process

In order to effectively develop the IDCT Tool, we investigated means to define and manage the system development process. Rational Unified Process (RUP), Unified Modeling Language (UML), and Rational Suite software were identified as methodologies and technologies that are well suited to a Web-based object-oriented IDCT Tool. The PI and a Masters student attended a training course on Rational Corporation’s Analyst Suite systems development software in Cupertino, CA. Aspects of the Rational Unified Process (RUP) and Rational’s Tool Suite were used to guide the development of the IDCT Tool.

A result of our investigation into traditional and object-based tools, was the publication of a paper entitled, “A Comparison of Systems Analysis Tools for Software Development,” authored by S. W. Ormon and A. G. Greenwood. The paper was presented at the Industrial Engineering Research Conference in Dallas in May 2001 and was published in their proceedings; a copy of the full paper is provided in Section 2.4, the Conference/Journal Articles & Technical Notes section of this report.

1.5.2.2 Development Environment

In order to begin demonstrating our concepts as quickly as possible, CRT-1 (Cost Risk Tool, version 1) was developed in *Visual Basic 6*.

Once we had a working prototype, we investigated alternative development environments for CRT-2. Our final choice was between *Java* and *VisualBasic.net*. *VB.net* was selected because Microsoft’s .net environment provides flexibility in terms of programming language, good database tools, and is object oriented. Also, the component software *KeyTreeView* was selected to create and manage the IDCT’s hierarchies. For CRT-3, we continue development in *VB.net* and plan to use Microsoft’s internal tree constructs.

1.5.2.3 CRT (Cost Risk Tool) Releases and Capabilities

CRT is the primary component of the IDCT Tool. Evolving prototypes of the Tool's design serve to illustrate needed capabilities and demonstrate proof of concept for meeting those needs, thereby employing an iterative design and development process approach.

CRT is further explained and illustrated in Section 1.6 and Section 3 (Briefing), as well as:

- Greenwood, A. G. and Ormon, S. W. "A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk," *Proceedings of the Decision Sciences Institute Conference*, November 2002, San Diego (full paper is included in Section 2.2)
- Chapter 3 of Ormon, Stephen W. "Development of a Hierarchical, Model-Based Design Decision-Support Tool for Assessing the Uncertainty of Cost Estimates," Masters Thesis, Department of Industrial Engineering, Mississippi State University, May 2002.

The following briefly describe the various releases/versions.

CRT-1 (completed December 2001)

- cost estimates are based on flexible and coupled product-breakdown structures (PBS) and cost-breakdown structures (CBS). The PBS is dynamic and the CBS is static (i.e., the CBS is embedded within each PBS component).
- includes simulation-based cost assessment in order to consider design variable and project parameter uncertainty; provides means to:
  - specify individual points of uncertainty
  - determine component-specific cost/risk uncertainty in terms of expected values, histogram, sensitivity analysis
  - assess relationships among components in terms of cost and risk (identify high cost/risk components) – scatter graph
  - cost/risk roll ups to subsystem and system levels
- model-centric cost assessment – a model base linked to CBS/PBS combination employing a simple model selection capability (really only includes the "look and feel" of the selection capability; the models are internal.)
- built upon relational database
- developed in *Visual Basic 6*.

CRT-2 (completed May 2002) – used to re-engage customers through an industry survey.

- dynamic CBS – user has the ability to create and modify CBS, i.e. it is not embedded within each PBS component.
- utilizes "Explorer" type tree interface for PBS and CBS
- PBS and CBS are coupled through a dynamic array-based structure.
- similar alternative designs are grouped into "packages"
- enhanced user interface
- incorporates data dictionary for variable definition and model registration
- incorporates facilities for registering models
- model selection enhancements:
  - models are external (initial linkage is to Excel, use of XML may follow)
  - based on data dictionary
- developed in *VB.net* – object oriented (class structure), easy to connect to other programs (XML support), cross Microsoft interfaces (C++ and C#), quick user interface development
- all flat data files

CRT-3 (nearly complete; self funded changes)

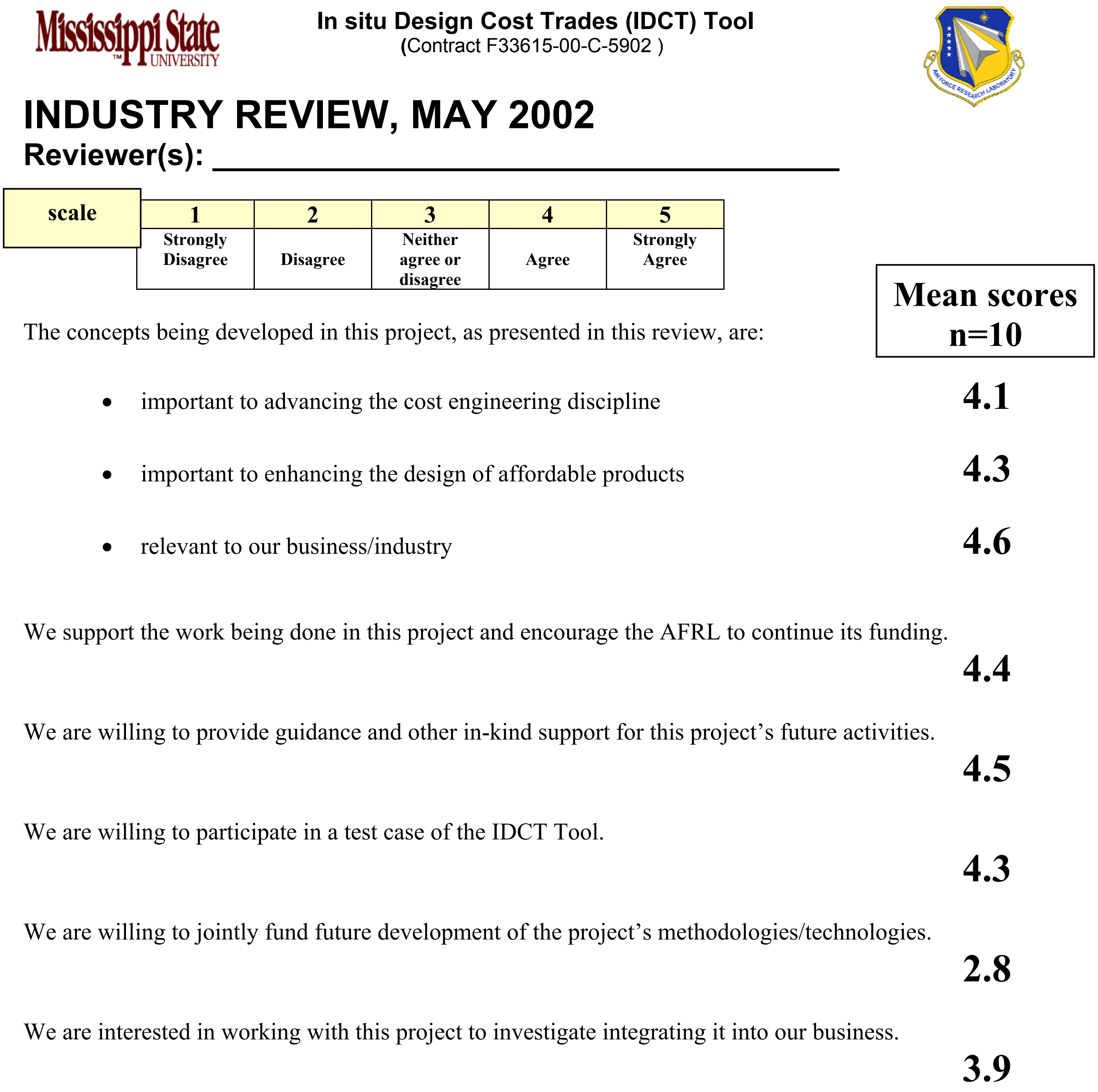
- completely redesigned to be truly object oriented to improve maintainability, stability, performance
- completely built upon relational database to improve maintainability, stability, performance, and reporting flexibility
- CBS elements can vary by PBS component
- estimate instance can incorporate multiple models (e.g. the product of a basic cost estimate, labor rate from table, inflation factor from table, and learning curve)
- simulations can be multithreaded
- “close” to server-based application
- improved user interfaces
- developed in *VB.net*

1.5.2.4 Industry Reviews

Since this project was oriented towards developing methodologies and technologies that were to be used in engineering practice, it was important to gain feedback from industry. During the proposal stage, the PI assembled a project support group composed of individuals with a variety of expertise and representing a wide range of perspectives. The group contained individuals from academia and industry with expertise in cost engineering and analysis (including high-level parametric analyses as well as detailed engineering build up; manufacturing, operations and support, and life-cycle; affordability assessment), product design (including the design process as well as engineering tools such as product data management and CAD), process analysis and design, systems engineering, and information systems. As the fundamental concepts and the overall approach for the IDCT Tool were being formulated in the early stages of the project, many from the project support group provided informal input to the project through discussions at conferences, meetings, and telephone conversations. This group was just beginning to be utilized on a more formal basis when the project was curtailed.

Just prior to the conclusion of the project, an industry survey was developed and administered in order to re-establish the industry contacts and identify potential industry partners to engage in further development of the IDCT Tool. The survey package included a PowerPoint presentation with notes on the project and prototype, two papers on portions of the project, and a questionnaire. The survey focused on key potential participants, i.e. “customers,” in order to obtain their feedback on the concepts and work performed to date, begin to develop ways to further engage potential users, and develop means to demonstrate the value of the project to the Air Force. Participants included: Acquisition Logistics Engineering (Charlie Coogan and Steve Rogers), Cognition Corporation (Brian Glauser and Mike Cronin), John Fondon (retired designer and systems engineer from Northrop Grumman), Frontier Technologies (Ron Shroder), Galorath Associates (Dan Galorath and Newlin Warden), General Electric Aircraft Engines (Mike Bailey), Pratt & Whitney (Joe Jaworski), SDRC (Mohsen Rezayat), TechniRep (Don Shrader), Tecolote Research (Harmon Withee, Peter Frederic).

The survey instrument that was developed and administered is provided as Figure 1-65. The combined results from the ten respondents are overlaid on the survey to the right of the question. As can be seen in Figure 1-6, the project was well received with the mean response for most questions being between Agree and Strongly Agree. In summary, the participants thought the project was relevant and important for advancing the field of cost engineering and the design of affordable products. The participants supported the work being done and encouraged AFRL to continue funding. They were also willing to provide guidance and support for future project activities, including participation in a test case.



In situ Design Cost Trades (IDCT) Tool
(Contract F33615-00-C-5902)

INDUSTRY REVIEW, MAY 2002
Reviewer(s): ______________________________

scale	1	2	3	4	5
	Strongly Disagree	Disagree	Neither agree or disagree	Agree	Strongly Agree

Mean scores
n=10

The concepts being developed in this project, as presented in this review, are:

- important to advancing the cost engineering discipline **4.1**
- important to enhancing the design of affordable products **4.3**
- relevant to our business/industry **4.6**

We support the work being done in this project and encourage the AFRL to continue its funding. **4.4**

We are willing to provide guidance and other in-kind support for this project's future activities. **4.5**

We are willing to participate in a test case of the IDCT Tool. **4.3**

We are willing to jointly fund future development of the project's methodologies/technologies. **2.8**

We are interested in working with this project to investigate integrating it into our business. **3.9**

Figure 1-6: Industry survey of project and Tool prototype.

As a result of the survey, we identified the most likely partners for further development to be:

- Galorath Associates, interface CRT with SEER H, SEER S, and SEER DFM, as well as leverage Galorath's CAD interface,
- either GEAE through the FIPER (Federated Intelligent Product EnviRonment) program,
- or more likely, Pratt & Whitney and Cognition Corporation, in order to utilize their Cognition's ECM (Enterprise Cost Management) product.

Unfortunately, these partnerships were not explored further due to AFRL's suspension of the project.

Two other formal interactions with members of the project support group included: (1) attendance at the 4th Annual Affordability and Cost Modeling Workshop, sponsored by AFRL in Fairborn, OH on 30-31 October 2001 in order to demonstrate and get feedback on preliminary concepts and earlier versions of the prototype; and, (2) a meeting with Don Shraeder of TechniRep to discuss possible collaboration through the A^3I (Advanced Aluminum Aerostructures Initiative). We discussed software that may help him assess life cycle cost and discussed ways in which this project might be demonstrated, evaluated, and tested at Boeing and Lockheed Martin.

1.5.3 Process Definition

The process definition activities of this project are presented after a brief discussion of specific aspects from the precursor study that provides motivation for this project. The processes that are addressed in this project include the design process, cost estimating process, and requirements engineering/management process. In addition, an approach to better represent these processes was developed and is descibed in the last portion of this section.

1.5.3.1 Extension of precursor work

One of the results of the precursor study was the definition of a high-level concept of operations for a generic IDCT Tool. This project extends the basic definition by exploring the activities on how the tool would be used in more detail. The concept of operations is extended in several ways. First, since design, cost evaluation, and requirements engineering/management are so tightly related, we explored each of these in more detail, including defining their activities and combining the activities into processes. Each of these efforts is discussed further in subsequent sections. Second, we defined the relationships between the activities within each process. This provides the basis for identifying the relationships between processes. These definitions -- both the activities and their relationships – provide the foundation for developing a tool that bridges the processes. The tool will not be effective unless there is a clear understanding of how the tool will be used and how it will enhance these critical processes.

An example of this concept is illustrated in Figure 1-7. The two shaded cloud-like symbols represent the design/IPPD decision-making and trade study process and the cost-evaluation process. Specific activities from the design/IPPD process that are directly related to cost evaluation are identified by the process flow diagram under the design/IPPD "cloud." Similarly, the cost-evaluation activities that are directly related to the design trade study process are identified by the process flow diagram under the cost-evaluation "cloud." The IDCT Tool provides the link between the two sets of processes. The heavy vertical arrows show the primary links and flows of information between the two sets of processes. These flows of information are managed by the IDCT Tool.

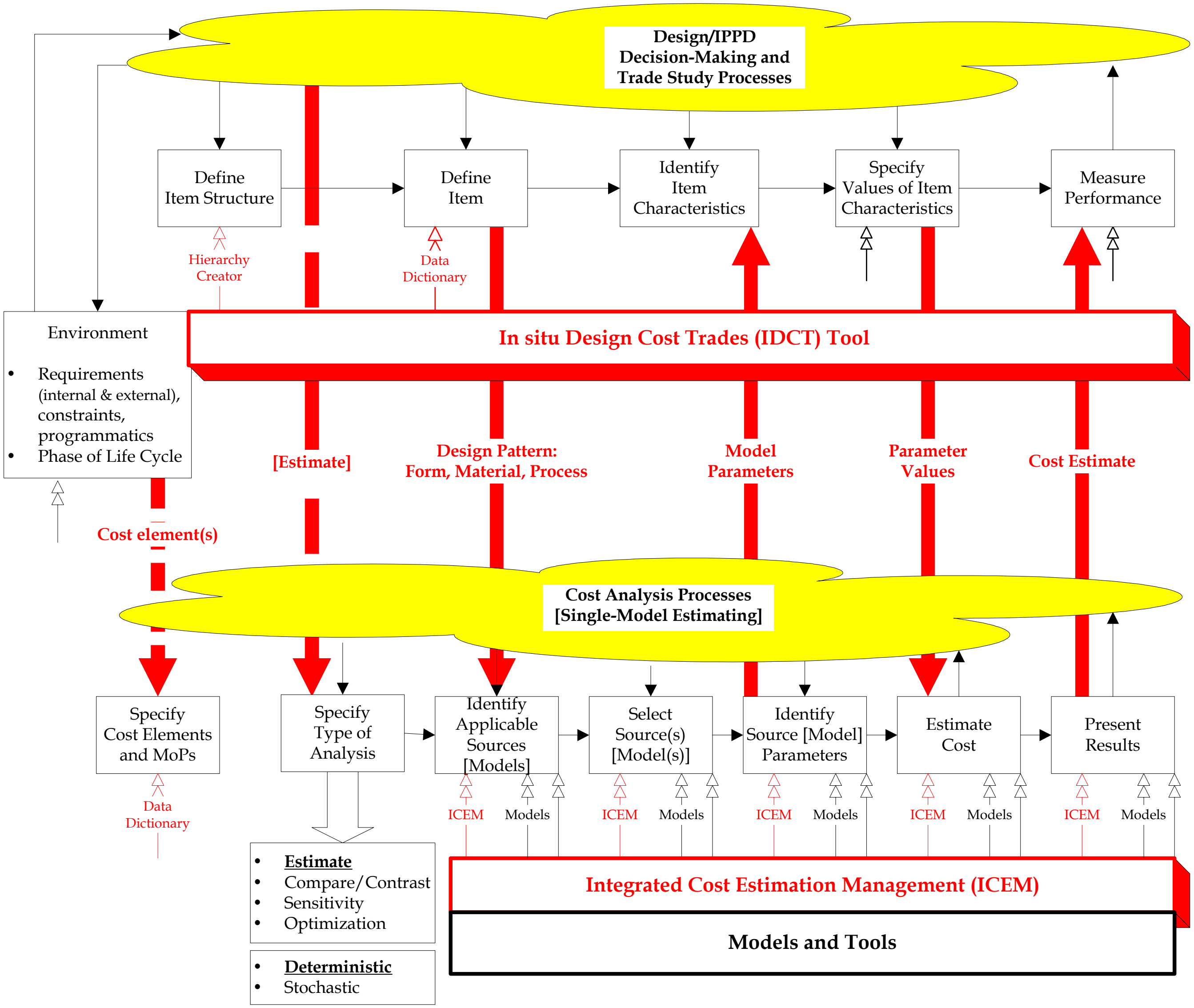


Figure 1-7: IDCT Tool links design and cost evaluation processes.

1.5.3.2 Definition of design activities and processes

A literature review was conducted in order to define the design process. Information from the literature was used to formulate an initial definition and representation of product design. This was reviewed by a MSU Aerospace Engineering faculty member to refine the representation. Subsequently, information from additional sources was incorporated. As part of this investigation, we recognized the need to identify an example to illustrate and test the design a process, as well as obtain industry input and feedback. We planned to do this but the project was curtailed before this activity was completed.

1.5.3.3 Definition of cost estimating activities and processes

In order to help identify ways to improve cost estimation, especially early in the product design process, and to facilitate the development of cost estimation tools, cost estimation needs to be viewed and represented as a process. Currently, there is no known documented generic cost estimation methodology that addresses the critical steps within the cost estimation process. The documented processes that do exist are industry or government organization specific. Also, the documentation does not use tools such as process modeling to effectively display the process in a coherent and efficient form. Two key activities in this project were to research cost estimation activities and from that information define a generic cost estimation process. The cost estimation processes for the Army, Air Force, Navy, and NASA were reviewed, described, and represented in a common format using the $IDEF_0$ methodology. These processes were then assimilated into a generic cost estimation process which is represented as an $IDEF_0$ diagram and each activity within the process is defined. The assimilated generic cost estimation process was extended and based on a separate investigation of risk/uncertainty methodologies. Details of this activity are provided in:

- Greenwood, A. G. and Ormon, S. W. "Development of a Generic Cost Estimation Process," completed; to be submitted for review to be presented at the *ASEM National Conference,* October 2003. (The full paper is included in Section II.3.)
- Chapter 2 of Ormon, Stephen W. "Development of a Hierarchical, Model-Based Design Decision-Support Tool for Assessing the Uncertainty of Cost Estimates," Masters Thesis, Department of Industrial Engineering, Mississippi State University, May 2002.

This activity is especially important because it provides the foundation for understanding the activities and tools that an IDCT Tool will need to support. It was planned to have the generic process reviewed by industry but the project was curtailed prior to obtaining the review.

1.5.3.4 Definition of requirements engineering/management activities and processes

Since defining and managing requirements is a critical aspect of the design process and closely linked to cost estimation, we performed a literature review to define the requirements process. Once the process was defined, we planned to link the requirements process with the design and cost estimation processes to create a comprehensive process representation upon which the IDCT Tool could be effectively designed.

The review of the requirements-related literature uncovered several key issues: there is confusion between the terms 'requirements engineering' and 'requirements management;' there is no underlying or unifying framework for requirements engineering/management; and, there is no methodology for effectively representing the process and activities involved in requirements. In order to address these issues, we:

(1) identified the similarities and differences between requirements engineering and requirements management and developed a comprehensive definition of each,

(2) developed a flexible, process-based framework for defining and managing requirements, and

(3) developed a tabular view and hybrid graphical view (HGV) for representing the requirements process. The HGV

It was planned to have the requirements engineering/management process reviewed by industry but the project was curtailed prior to obtaining the review.

Details of this activity are provided in:

- Greenwood, A. G. and Liaw, J. A. "A Framework for Engineering and Managing Requirements," in preparation for submission to *IEEE Transactions on Engineering Management*. (The full paper is included in Section 2.5.)
- Liaw, Judy. "Definition and Representation of Requirements Engineering/Management: A Process-Based Approach," Masters Thesis, Department of Industrial Engineering, Mississippi State University, May 2002.

1.5.3.5 Means to represent processes

As described above, we developed a hybrid process representation scheme (an assimilation of IDEF$_0$, IDEF$_3$, and cross-functional process mapping) to provide a richer process representation. Details of this activity are provided in:

- Section 4.2.2 in Greenwood, A. G. and Liaw, J. A. "A Framework for Engineering and Managing Requirements," in preparation for submission to *IEEE Transactions on Engineering Management*. (included in Section 2.5)
- Chapter 3 in Liaw, Judy. "Definition and Representation of Requirements Engineering/Management: A Process-Based Approach," Masters Thesis, Department of Industrial Engineering, Mississippi State University, May 2002.

Since dealing with processes is a significant component of the project, we investigated means to manage process data and representations. Phios was identified as a potentially viable technology to facilitate organizing, navigating, and maintaining interconnected networks of processes, activities, and process knowledge. It is Web-based but can be used standalone. We signed a license agreement with Phios Corporation of Cambridge, MA for use of their Phios Process Repository software (including the server, process editor, and viewer), completed installation, and conducted an initial investigation of the software. We concluded, based on the initial investigation that the software did not appear to have the capability that we expected. We suspended further investigation until we could attend a Phios training program and discuss our needs with them one-on-one. The project was curtailed before we could complete our investigation.

1.5.4 Investigation of Supporting Technologies

In order to develop a comprehensive and effective IDCT Tool, the following supporting technologies were investigated, in addition to those discussed above: engineering design tools, simulation-based design, model management, AIMMS development environment computational web portals, manufacturing cost modeling, operations and support cost modeling, reliability models, general capabilities needed for cost assessment, and web-based collaborative software.

1.5.4.1 Engineering Design Tools

Initial investigations were conducted into the roles of CAD, STEP, and PDM technologies in the operation of the IDCT Tool.

1.5.4.2 Simulation Based Design

Simulation based design (SBD), also referred to as virtual prototyping, was identified as an important evolving area that is very applicable to the development and use of an IDCT Tool. Virtual prototypes play a very active role in the new product development initiatives. With the aid of virtual prototypes and CAD/CAM tools, products can be produced with a more robust design and shorter manufacturing cycle time. Many major manufacturers use SBD.

Simulation-based design refers the computer-based simulation of a system or subsystem, with a degree of functional realism comparable to a physical prototype, in order to addresses the engineering design concerns of the developer, the process concerns of the manufacturer, the logistical concerns of the planner, and the training and programmatic concerns of the operator. A summary of the literature review is provided as a technical note in Section 2.7.

1.5.4.3 Model Management

Since the IDCT Tool makes heavy use of a variety of types of models, model management is a critical need of the Tool. Therefore, we conducted an initial literature review on model management technologies. A summary of that initial investigation is provided in the Technical Note in Section 2.8.

Two critical methodologies/technologies were identified for further exploration: Structured Modeling, which provides a formal mathematical framework and computer-based environment for conceiving, representing, and manipulating a wide variety of models (proposed by Geoffrion in 1987) and XML – Extensible Markup Language—which is used to build highly extensible and interoperable software solutions for the Web.

1.5.4.4 AIMMS

Since the IDCT Tool is a computationally-intense, model-driven decision-support system, we investigated AIMMS (Advanced Integrated Multidimensional Modeling System) software from Paragon Decision Technologies. AIMMS provides a development environment for interactive decision support systems that use advanced computational techniques. We installed a trial version, reviewed documentation, and began to develop simple examples to better understand and test capabilities. Based on this investigation, we decided that *AIMMS*, in its current form, is not directly applicable to the development of the IDCT Tool. *AIMMS* is a decision-support-system development environment primarily for mathematical programming models. However, *AIMMS* is being adopted as part of another project being conducted within our Lab; therefore, we continued to learn about its capabilities and applicability to cost models and the IDCT Tool. Also, further interactions with the company, Paragon Technology, could help us identify applications to future versions of the IDCT Tool.

1.5.4.5 Computational web portals

Computational web portal technology is being developed at MSU’s NSF Engineering Research Center for Computational Systems. This technology will provide web access to remote computational resources (hardware, software, and data) and hide the complexity of heterogeneous and distributed computing environments. This technology is potentially applicable to advanced versions of the CRT and more comprehensive engineering design-support systems. We conducted a preliminary investigation of the technology and periodically monitored its evolution so that it could be employed in future versions of the CRT.

Attended two webinars (web seminars): (1) integrating structured and unstructured data through portals and (2) XQuery to query XML trees. Both provided an introduction to technologies that may be applicable to the IDCT Tool.

1.5.4.6 Manufacturing cost models

In order to demonstrate the concepts developed in this project, as implemented in the IDCT Tool prototypes, and to test the operation and performance of the prototypes, we researched and investigated cost models to link to the prototypes. In the early phases of development, our focus was on the basic architecture of the Tool and not on sophisticated links to distributed, disparate costing systems with proprietary interfaces. (This would be a later capability, possibly realized through the use of the evolving computational web portal technology). While numerous available models and costing systems were investigated, the following two were considered the most promising.

1.5.4.6.1 TAPSI

The TASPI (Tool for Aircraft Structures based on Process Information) software is based on the preliminary design methodology that was developed by MSU's Dr. Rais-Rohani as part of a grant from NASA Langley in 1998. We attempted to isolate and extract the cost models from the remainder of the TASPI code and recompile them as separate modules that could be easily interfaced with the CRT. The extraction of computer code was suspended because the code was poorly documented and the modules were so commingled that the cost-relevant portions could not be isolated. The problems that were encountered in this activity were documented.

1.5.4.6.2 RAND models in MS Excel

We identified some very simple cost estimating relationships from the following sources:

- Raymer, D.P., *Aircraft Design: A Conceptual Approach*, 3rd Ed., American Institute of Aeronautics and Astronautics, Reston VA., 1999.
- Restar, S.A., Rogers, J.C., and Ronald, W.H., *Advanced Airframe Structural Materials, A Primer and Cost Estimating Methodology*, RAND Corporation Report R-4016-AF.

These relationships, provided in Appendix B, were coded in MS Excel as separate models. They served their intended purpose of demonstrating CRT's concept for registering and managing models. We planned to expand the model base to include a wider variety of models and modeling types but the project was curtailed prior to undertaking the model expansion activities.

1.5.4.7 Operations and Support Cost Modeling

Since most of our focus in the foundation study was on manufacturing cost and since a major project goal is to have the IDCT Tool be a life-cycle cost tool, we investigated operations and support cost estimating models, methodologies, and technologies.

The PI and a student attended the MAAP (Monterey Activity-based Analysis Platform) Forum in June 2001 in Melbourne, FL. The forum was for developers, users, and potential users to understand the full capabilities of the latest version of MAAP and provide directions for future development. MAAP is total ownership cost modeling and analysis tool. It is event-driven rather parametric; i.e., cost estimates are generated by aggregating resources predicted to be consumed by the activities of the system. MAAP is a derivative of EDCAS (Equipment Designer's Cost Analysis System), a leading front-end analysis tool for life cycle cost and logistics modeling early in the design process. The developers of MAAP and EDCAS, Systems Exchange, and their customers were considered potential partners for this project.

As a result of the conference we began investigating three life-cycle cost modeling and estimating tools -- MAAP and EDCAS, both from Systems Exchange, and ASCET (Aircraft System Cost Estimating Tool), from Tecolote Research. The focus of the analysis was on the operations and support (O&S) component of life-cycle cost. As indicated earlier, MAAP and EDCAS take an activity approach to generating O&S cost; whereas, ASCET is primarily parametric based. These software were investigated not only to define the interface requirements for specific links to the IDCT Tool but to identify general capabilities that are needed in any cost environment and to identify strengths and weaknesses of these tools when applied early in the design process.

1.5.4.8 Reliability models

A major driver of operations and support cost is reliability. Since it is a project goal to have the IDCT Tool be a life-cycle cost tool and since most of our focus in the foundation study was on manufacturing cost, we deemed it necessary to investigate the use of reliability models early in the design process.

We found that during the early stages of conceptual design, the ability to predict reliability is very limited. Most of the models that are used are extremely specific to an individual system or industry. We identified five general procedures (using both simulation and analytic solution techniques) for predicting system reliability and average mission cost. The procedures consider both known and unknown failure rates and component- and subsystem-level analyses. The estimates are based on the number of series subsystems and redundant (active or stand-by) components for each subsystem. Software was developed, referred to as RPM (Reliability Prediction Models), that facilitates the application of the simulation-based techniques. More information on this work is provided in:

- Ormon, S. W., Cassady, C. R., and Greenwood, A. G. "Reliability Prediction Models to Support Conceptual Design," *IEEE Transactions on Reliability*, vol. 51, no. 2, June 2002, pp. 151-157. (The full paper is included in Section 2.1.)
- Ormon, S. W., Cassady, C. R., and Greenwood, A. G. "A Simulation–Based Reliability Prediction Model for Conceptual Design," *2001 Proceedings of the Annual Reliability and Maintainability Symposium*, January 2001, Philadelphia, pp. 433-436. Note this paper was the winner of the *The Society of Reliability Engineers' Stan Ofsthun Best Paper Award.*

1.5.4.9 Capabilities required to support cost analyses

In support of the effort to develop the IDCT Tool, we began developing a broad list of capabilities that are needed in order to effectively perform cost analyses and identify the means by which these capabilities are being met. This list, derived from the literature and investigation of cost-analysis software, was intended to result in a framework of capabilities. This effort was carried out in parallel with defining the cost estimation, design, and requirements processes. Once both tasks were completed, the plan was to update the capabilities list with information derived from the process definitions. Finally, the elements of the capabilities framework were to be prioritized, partly through an industry survey. This framework would provide the basis for the IDCT Tool development and the development of other cost-related technologies.

A preliminary and high-level list of capabilities along with their status at each phases of a product's life cycle is presented in Appendix 1-C. A more detailed list of capabilities that were gleaned from the literature to date is provided as Appendix 1-D.

1.5.4.10 Web-based collaboration software services

In conjunction with another project, we investigated Web-based collaboration software services. We selected *WebEx* for a pilot project and eventually adopted it as our collaboration software service. *WebEx* allows flexible and spontaneous sharing of content and applications along with audio and video conferencing; sessions/meetings may be recorded, posted to a website, and viewed by those unable to attend the meeting. It is a hosted service; i.e., all that is needed is a Web browser and telephone, thus eliminating the need for investment in equipment, software installation, training, and maintenance. The importance of this technology to this project was:

- basic understanding of collaboration technologies since the IDCT Tool will eventually need to function in a Web-based collaborative environment and may need to interface with this type of technology,
- facilitate communications with the AFPM and project participants in industry and academe, and
- facilitate software installation and demonstration through *WebEx*'s applications sharing and remote desktop control.

1.5.5 Dissemination: Presentations and Publications

The following is a summary of presentations and publications directly related to this research. Most of the papers are included in Section II.

1.5.5.1 Journal articles published:

Ormon, S. W., Cassady, C. R., and Greenwood, A. G. "Reliability Prediction Models to Support Conceptual Design," *IEEE Transactions on Reliability*, vol. 51, no. 2, June 2002, pp. 151-157. The article is provided in Section 2.1.

1.5.5.2 Conference papers completed/published:

Ormon, S. W., C. R. Cassady, and Greenwood, A. G. "A Simulation–Based Reliability Prediction Model for Conceptual Design," *2001 Proceedings of the Annual Reliability and Maintainability Symposium*, January 2001, Philadelphia, pp. 433-436. Note this paper was the winner of the *The Society of Reliability Engineers' Stan Ofsthun Best Paper Award*.

Ormon, S. W., and Greenwood, A. G. "A Comparison of Traditional and Object-Oriented Systems Analysis Tools," *10th Annual Industrial Engineering Research Conference (IERC)*, May 2001, Dallas. The article is provided in Section 2.4.

Greenwood, A. G. and Ormon, S. W. "A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk," *Proceedings of the Decision Sciences Institute Conference*, November 2002, San Diego. The article is provided in Section 2.2.

Greenwood, A. G. and Ormon, S. W. "Development of a Generic Cost Estimation Process," completed; to be submitted for review to be presented at the *ASEM National Conference,* October 2003. The article is provided in Section 2.3.

1.5.5.3 Journal articles in preparation:

Greenwood, A. G. and Liaw, J. A. "A Framework for Engineering and Managing Requirements," in preparation for submission to *IEEE Transactions on Engineering Management*. The article is provided in Section 2.5.

1.5.5.4 Others presentations, demonstrations, and publications

The PI made several presentations about this project to representatives from, among others, Nissan, National Science Foundation, and the National Automotive Center. These presentations were made in conjunction with the new Center for Advanced Vehicular Systems (CAVS) at MSU that is part of the University's support of Nissan's major assembly plant that is being built in Mississippi. These meetings were intended to present MSU's capabilities and identify opportunities for collaborative research.

Initial work began on a project website and the first release was nearly complete by the time work was suspended.

Attended the AFRL Cost Workshop (October 2001) to better understand the Air Force's needs in the area of cost assessment, get an update on the current state of the art in cost and affordability modeling, and identify potential industry support in terms of project input, review, and evaluation. Some of the key "take aways" from the conference that are related to this project include:

- The conference reinforced the PI's belief that there are many useful tools available and being developed in the cost/affordability arena; however, there is no "strategic plan" and/or "roadmap" for how they fit together, no "catalogue" of what alternatives and capabilities are available to analysts, and no assessment of capability "gaps" that need to be addressed.
- AFTOC is a mature information system that is being woven into Air Force decision making. Methodologies/tools need to be developed to use the system effectively in order to identify cost drivers for use in early design decision making.
- Cost methodologies and tools are needed that can assess the impact of change -- e.g. technology refreshment, obsolescence – so that engineers can make meaningful trades as they design for change.
- Process imbues credibility which reinforces our efforts to define and represent a generic cost analysis process and define the capabilities that are needed to effectively carryout the process.
- Cost tools need to be linked to design tools.

The RPM (Reliability Prediction Model) and CRT (Cost Risk Tool) prototypes were demonstrated at the AFRL Cost Workshop. Even though these were early prototypes several comments are noteworthy.

- It is important that the IDCT Tool be integrated with CAD tools. The PI is aware of this issue but feels we need a better prototype in order to illustrate our concepts before actively engaging the CAD vendors.
- IDCT Tool needs to leverage existing risk assessment tools, e.g. those within SEER, Crystal Ball, etc.
- The prototypes need to have a more commercial look and feel. Again, we were testing concepts as quickly as possible in order to identify the capabilities of the IDCT Tool before

committing resources to make the tools behave and look more professional. Also, we explored higher-level development environments, e.g. AIMMS, to facilitate development.

Although there was no direct impact on the project, the PI offered a special topics course in Affordability Analysis to a Ph.D. student at Lockheed Martin (Fort Worth) through MSU's distance education program. This was intended to provide a contact within the company in order to get information and feedback on our approach, methodologies, tools, etc.

1.6 Cost Risk Tool

This section briefly describes the latest version of the CRT Tool, including a summary of the Tool's capabilities, definition of its foundation, definition of its architecture and systems design, discussion of its operation and interfaces, and ideas for future development. Additional information is included in the following paper which is provided as Section 2.2:

> Greenwood, A. G. and Ormon, S. W. "A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk," *Proceedings of the Decision Sciences Institute Conference*, November 2002, San Diego.

1.6.1 Capabilities

As has been noted earlier, the CRT prototypes provide demonstrations of the basic concepts developed in this research and the fundamental capabilities that we believe should be the foundation of all cost estimates. The following is a list of capabilities and features that are either currently included in the CRT or are a part of the current design and are expected to be incorporated in a near-term release.

- based on design and cost-analysis processes; interface defined through use cases
- simple robust model-centric architecture
  - object-oriented (developed in VB.net)
  - utilizes data dictionary
  - built on database; close to client/server
  - related alternatives are "packaged" as projects
  - incorporates model registration and management
  - encourages the use of variable-complexity models
- flexible, coupled, dynamic, hierarchical product- and cost-breakdown structures
- considers the impact of design-variable uncertainty on component and system cost (via an integrated Monte Carlo simulation engine)
- facilitates risk analysis (output displays)

1.6.2 Foundation

The CRT supports the basic cost estimation operations that are defined in the following representation:

$$Cost_{Sys} = \sum_{j=1}^{n} \left[\sum_{k=1}^{s} M_k^{*} \left(X_1^k, \ldots, X_l^k, \ldots, X_p^k \right) \right] \quad \text{where:}$$

- $M_k^* \in \left\{ M_k^1, M_k^2 M_k^3, \ldots \right\}$ and

- each X_l^k is either deterministic or X_l^k ~ *Triangular* (min, mode, max).

Cost$_{sys}$ is the total cost of a system that is being designed. It is an aggregate of estimates made at the subsystem or *component* level; i.e., as shown in the relationship above, it is the sum of the total cost of *n* components. Each component-level estimate is an aggregate of different types of cost *elements*, e.g. manufacturing labor, material, engineering overhead; i.e., as shown in the relationship above, it is the sum of *s* cost elements. Therefore, in order to estimate the cost of a component *j*, the cost of each cost element *k* is estimated using some type of model and based on the characteristics of component *j*. That is, the model transforms component characteristics into a cost. The model is represented in the above relationship as $M_k^*\left(X_1^k,...,X_l^k,...,X_p^k\right)$, where M_k^* denotes the selected model that is used to estimate cost element *k* and $X_1^k,...,X_l^k,...,X_p^k$ are the *p* component characteristics or variables that are used to estimate the cost element. The * superscript in M_k^* indicates that the model has been selected from some set of candidate models, $\{M_k^1, M_k^2 M_k^3,...\}$, that can estimate cost element *k*. The selection of a model is based on the type and amount of information that is currently available; i.e., it depends on the maturity of the component's design.

The above representation is implemented in the CRT as follows. A system is composed of *n* components and is represented as a hierarchical tree structure that we refer to the *PBS* (product breakdown structure). The cost of the system and each component is estimated based on a set of *s* cost elements; the structure is represented as a hierarchical tree structure that we refer to as the *CBS* (cost breakdown structure). The cost estimate for element *k* for component *j* is referred to as a *cost instance*; the notion of a cost instance is explained further below. In order to estimate the value of a cost instance, a model, M_k^*, is selected from a set of candidate models (the set of available models is referred to as a model base). Each model uses characteristics of the system -- component- and system-level parameters and variables, X_l^k -- to estimate cost element *k* for component *j*. The value of each variable may be either deterministic or stochastic. Currently within CRT, if a variable is identified as stochastic, we assume the values to be triangularly distributed and the user provides estimates for the three parameters of triangular distribution: minimum, most likely, and maximum values. The cost estimates for each component is the sum of its cost elements, as defined in the CBS; the system and subsystem cost estimates are the sum of its components, as defined by the PBS.

Note, that while there is one overall or meta cost structure for the system, the level of detail can vary by component. The meta structure is a template and defines the cost elements at the lowest or most detailed level. The estimates made at the component level may be made at any level and include only a subset of the elements. The level of the estimate will depend on the approach that is used, e.g. estimates using parametric models will be made at a higher level than estimates that are constructed from a detailed buildup. The approach used, and subsequently the model that is used, to provide an estimate depends on the amount of information known about the component; i.e., the CRT encourages the use of variable complexity models where the cost of well-defined components are based on detailed buildups, or even actual costs, and components that are based on evolving technologies or are not clearly defined yet in the design process are based on high-level parametric cost estimating relationships.

In developing the CRT, we consider two types of cost models:

1) *primary* models that estimate the cost with a specific cost element based on attributes of a component. Such models are typically in the form of cost estimating relationships (CER) that are available either from local functions or through commercial packages (e.g. PRICE, SEER). The model may also involve direct data that is obtained from a simple lookup (production or field

data). Unless the component is "off-the-shelf," the data are often modified based differing component characteristics; in this case, the model would be considered a CER.

2) *adjustment* models that are applied to the results of the primary models in order to:
   a) compensate for differing production quantities (learning) or year dollars
   b) convert model results, e.g. from hours to dollars, from direct cost to total cost

Currently, CRT only employs primary models. The next release, CRT-3, will include adjustment models.

As introduced above, the fundamental building block in the CRT is the estimate instance; it is represented in Figure 1-8 in IDEF$_0$ format. The cost instance links the type of cost that is being estimated (a control) with product/process characteristics (inputs that are transformed into output – cost estimate). Models are directly associated with each component cost-element combination; in IDEF$_0$ terms, they are mechanisms. Models are a primary aspect of cost analyses since they provide the means to transform some combination of design variables, program parameters, and product, process, and operations characteristics into estimates of system cost. Another mechanism is a means to assess the risk associated with the estimate, e.g., obtained through Monte Carlo simulation. Therefore, in order to produce an estimate of cost risk, the CBS, PBS, and model base must be coupled or linked together and with a simulation engine.

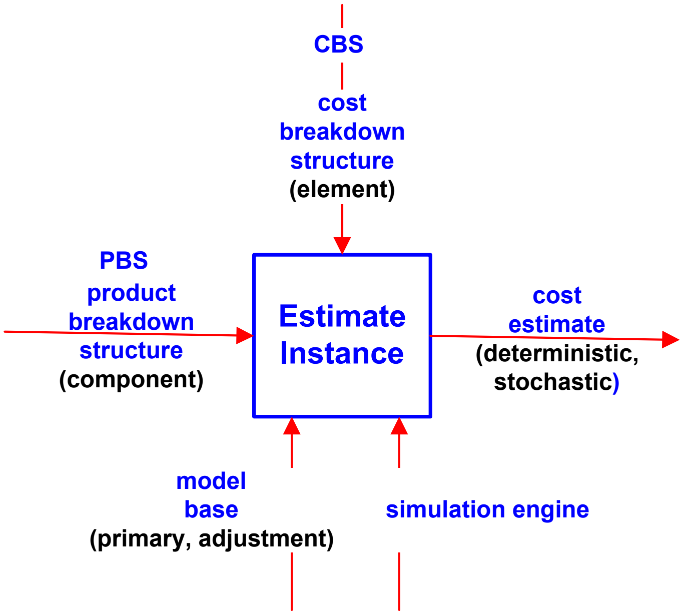
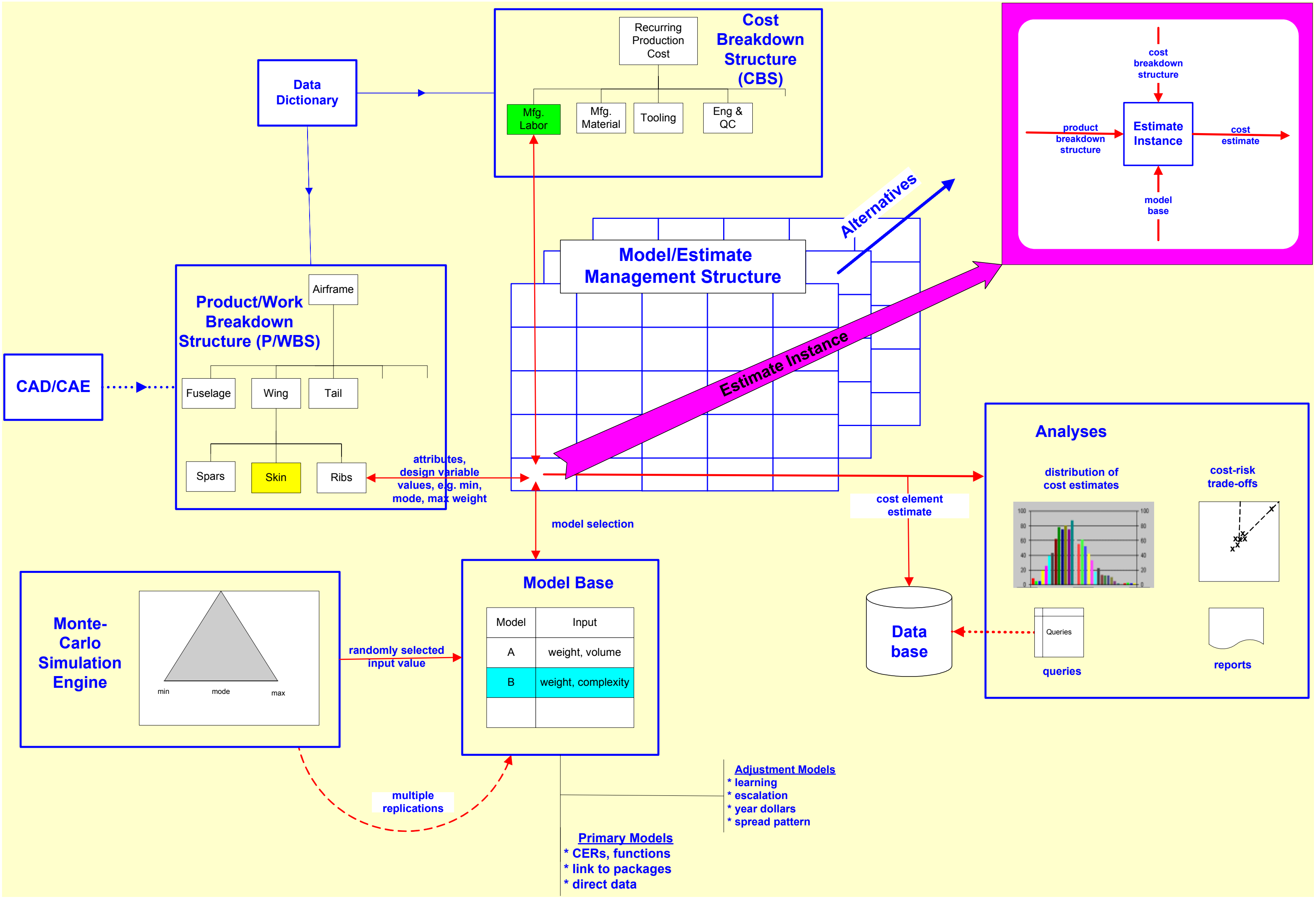


Figure 1-8: IDEF$_0$ representation of an estimate instance.

The set of all estimate instances for an alternative forms an array structure, termed the *Model/Estimate Management Structure*(MEMS). CRT creates a separate MEMS for each alternative and groups alternatives into projects. The estimate instance and MEMS provide the core of CRT's architecture, as defined in the next section.

1.6.3 Architecture and Systems Design

The estimate instance is the basic building block in our approach to cost estimating, therefore, the CRT is built around the estimate instance notion and is designed to provide the functionality needed to implement and manage estimate instances. Figure 1-9 provides the overall architecture for the CRT. It is composed of several interrelated modules that are directly linked to the estimate instance.


Data Dictionary
Cost Breakdown Structure (CBS)
Recurring Production Cost
Mfg. Labor
Mfg. Material
Tooling
Eng & QC
cost breakdown structure
product breakdown structure
Estimate Instance
cost estimate
model base
Alternatives
Model/Estimate Management Structure
Estimate Instance
CAD/CAE
Product/Work Breakdown Structure (P/WBS)
Airframe
Fuselage
Wing
Tail
Spars
Skin
Ribs
attributes, design variable values, e.g. min, mode, max weight
model selection
cost element estimate
Analyses
distribution of cost estimates
cost-risk trade-offs
Data base
Queries
queries
reports
Monte-Carlo Simulation Engine
min
mode
max
randomly selected input value
Model Base

Model	Input
A	weight, volume
B	weight, complexity

Figure 1-9: Overall architecture of the CRT.

As shown in Figure 1-9, the CRT is built around the array-based MEMS; each cell in MEMS represents an estimate instance. The rows in the array correspond to components in the PBS and columns in the array correspond to elements in the CBS. Most of the inputs to a cost instance are values of the attributes for components of the PBS, i.e., characteristics of the product being developed. In IDEF terminology, the controls of a cost instance are derived from the elements of the CBS. Therefore, two modules within CRT manage the PBS and CBS, as shown in the blocks to the left of and above the MEMS in Figure 1-9. Also shown in Figure 1-9 is the linkage of these two blocks through a Data Dictionary block. Each PBS and CBS is built and maintained through a data dictionary in order to provide consistency.

Again in IDEF terminology, the main mechanism of a cost instance, or means to generate a cost estimate, is the model base. Both primary models and adjustment models are retained in the model base. An important capability connected to the model base in CRT is the Monte Carlo Simulation Engine. It provides the ability to assess the impact of uncertainty in design variables on the cost of the system. Finally, there is a module that handles the output of a cost instance and provides analysis capability to assess cost risk, identify cost/risk drivers, provide reports and graphs, etc.

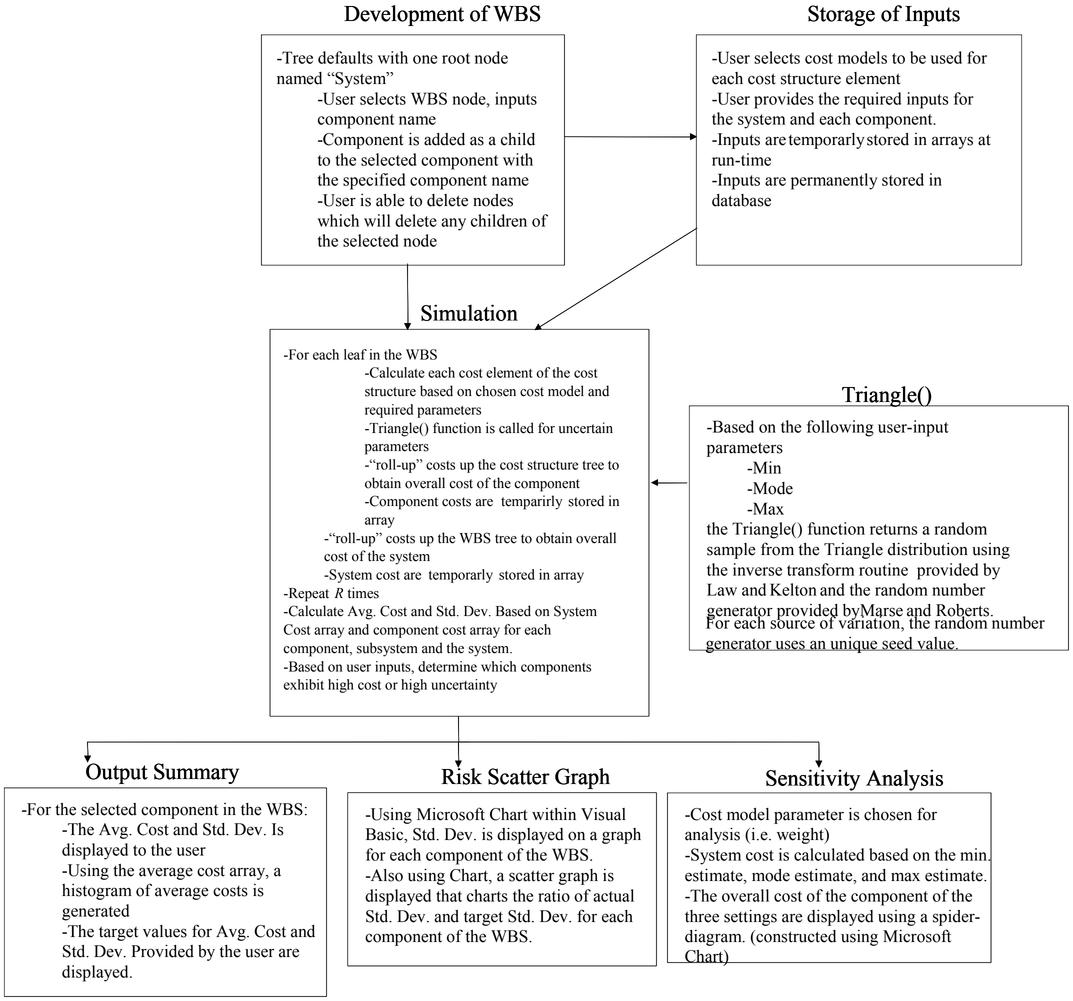
In order to define how CRT's modules operate and interact with other modules, a flowchart of the first version of CRT (CRT-1) is provided in Figure 1-10. CRT-1 is simpler than the current version in that the CBS is static (i.e., it remains the same for each component) and there is no model base (i.e., all components use the same estimating model). However, the simplicity provides a clearer illustration of the operations within and the basic relationships between the PBS or input module, the simulation engine, and the output modules.

In order to further define how CRT operates two use diagrams, that correspond to Versions 2 and 3, are provided as Appendix 1-B and 1-C, respectively. Use case diagrams describe how a system is used; i.e., they show the functionality of the system. The diagrams in the appendix define the purpose of each of CRT's functions and the sequence of actions that are performed to carry out the function.

Another means to define CRT is through an Entity-Relationship (ER) diagram. It graphically represents a system's entities (represented as boxes including entity attributes and methods) and their relationships (represented as diamonds).The ER diagram for CRT-3 is provided in Figure 1-11.

Table 1-1 provides a summary of the evolution of CRT in terms of its capabilities in the various releases/versions. The first version of CRT was completed in December 2001, the second version in May 2002, and CRT-3 is nearly complete, in terms of the capabilities shown in Table 1-1. The intent throughout the project has been for CRT to be object-oriented and relational database driven; CRT-3 most fully meets those objectives. All versions have been developed in Visual Basic (VB), with the latest two in VB.net. CRT has always been based on the estimate instance concept that was defined earlier. The array-based MEMS architecture was introduced in CRT-2. The two hierarchical structures, the PBS and CBS, have evolved with each version. Since models are a key aspect of cost analyses, it is important to control the models that are used in the analyses and to be able to effectively select models to be applied to each estimate instance. Both model registration and model selection capabilities, in conjunction with the system's data dictionary, were implemented in CRT-2.

As shown in Table 1-1, the simulation engine has remained virtually the same since the first release. It incorporates a well proven means to generate random samples from a triangular distribution and employs the variance reduction technique referred to as Common Random Numbers. CRT makes it easy for the user to specify from the PBS the variables that contain uncertainty and thus that need to be simulated. The simulation provides multiple measures of uncertainty/risk including multiple risk assessment graphs and "roll ups" of component risk to the system level.



Development of WBS

-Tree defaults with one root node named “System”

- -User selects WBS node, inputs component name
- -Component is added as a child to the selected component with the specified component name
- -User is able to delete nodes which will delete any children of the selected node

Storage of Inputs

-User selects cost models to be used for each cost structure element
-User provides the required inputs for the system and each component.
-Inputs are temporarly stored in arrays at run-time
-Inputs are permanently stored in database

Simulation

-For each leaf in the WBS

- -Calculate each cost element of the cost structure based on chosen cost model and required parameters
- -Triangle() function is called for uncertain parameters
- -“roll-up” costs up the cost structure tree to obtain overall cost of the component
- -Component costs are temparirly stored in array

-“roll-up” costs up the WBS tree to obtain overall cost of the system
-System cost are temporarly stored in array

-Repeat *R* times
-Calculate Avg. Cost and Std. Dev. Based on System Cost array and component cost array for each component, subsystem and the system.
-Based on user inputs, determine which components exhibit high cost or high uncertainty

Triangle()

-Based on the following user-input parameters

- -Min
- -Mode
- -Max

the Triangle() function returns a random sample from the Triangle distribution using the inverse transform routine provided by Law and Kelton and the random number generator provided byMarse and Roberts.
For each source of variation, the random number generator uses an unique seed value.

Output Summary

-For the selected component in the WBS:

- -The Avg. Cost and Std. Dev. Is displayed to the user
- -Using the average cost array, a histogram of average costs is generated
- -The target values for Avg. Cost and Std. Dev. Provided by the user are displayed.

Risk Scatter Graph

-Using Microsoft Chart within Visual Basic, Std. Dev. is displayed on a graph for each component of the WBS.
-Also using Chart, a scatter graph is displayed that charts the ratio of actual Std. Dev. and target Std. Dev. for each component of the WBS.

Sensitivity Analysis

-Cost model parameter is chosen for analysis (i.e. weight)
-System cost is calculated based on the min. estimate, mode estimate, and max estimate.
-The overall cost of the component of the three settings are displayed using a spider-diagram. (constructed using Microsoft Chart)

Figure 1-10: Relationships among CRT-1’s input, output, and simulation engine.

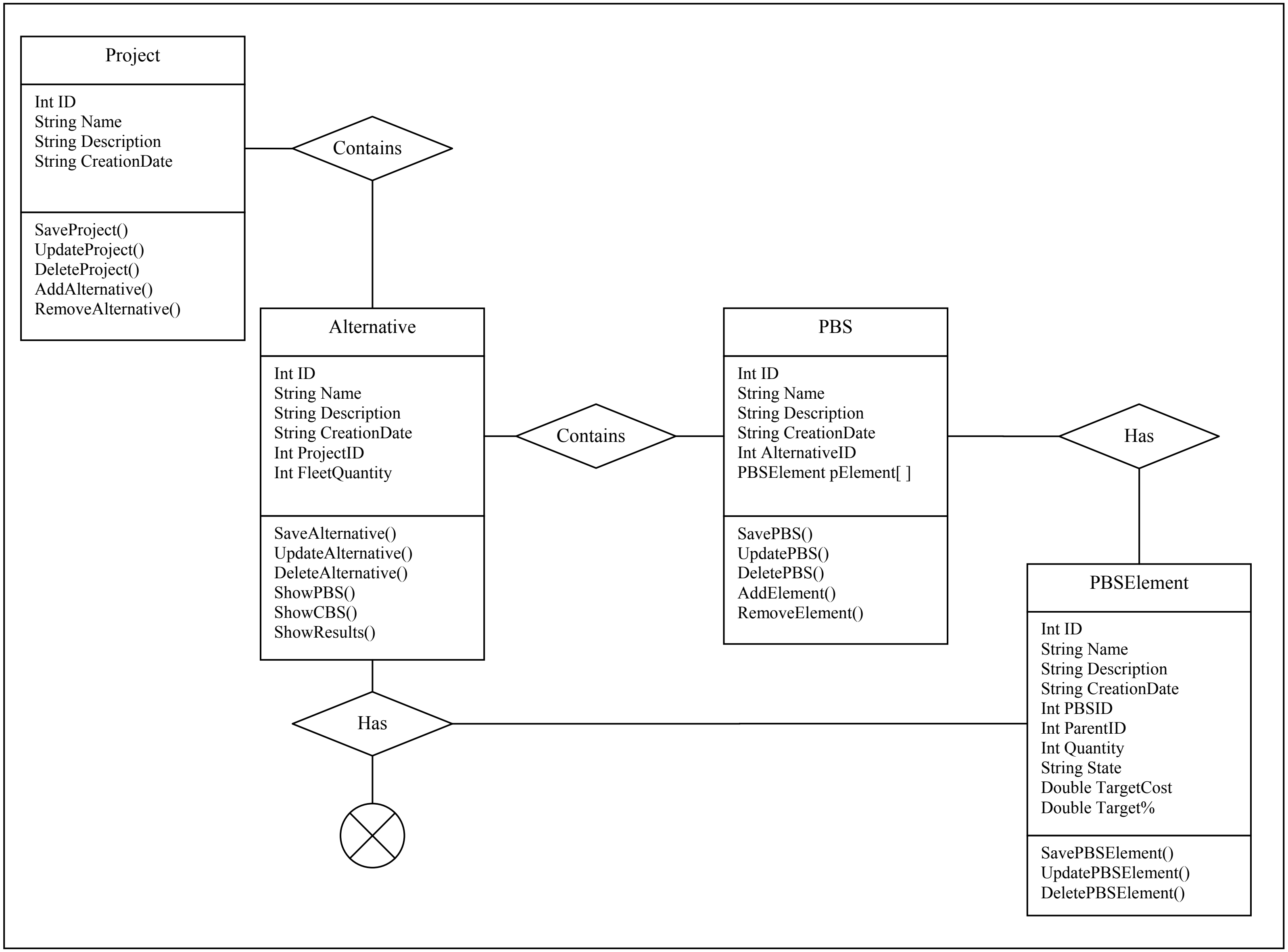


Figure 1-11: Entity Relationship Diagram for CRT-3.

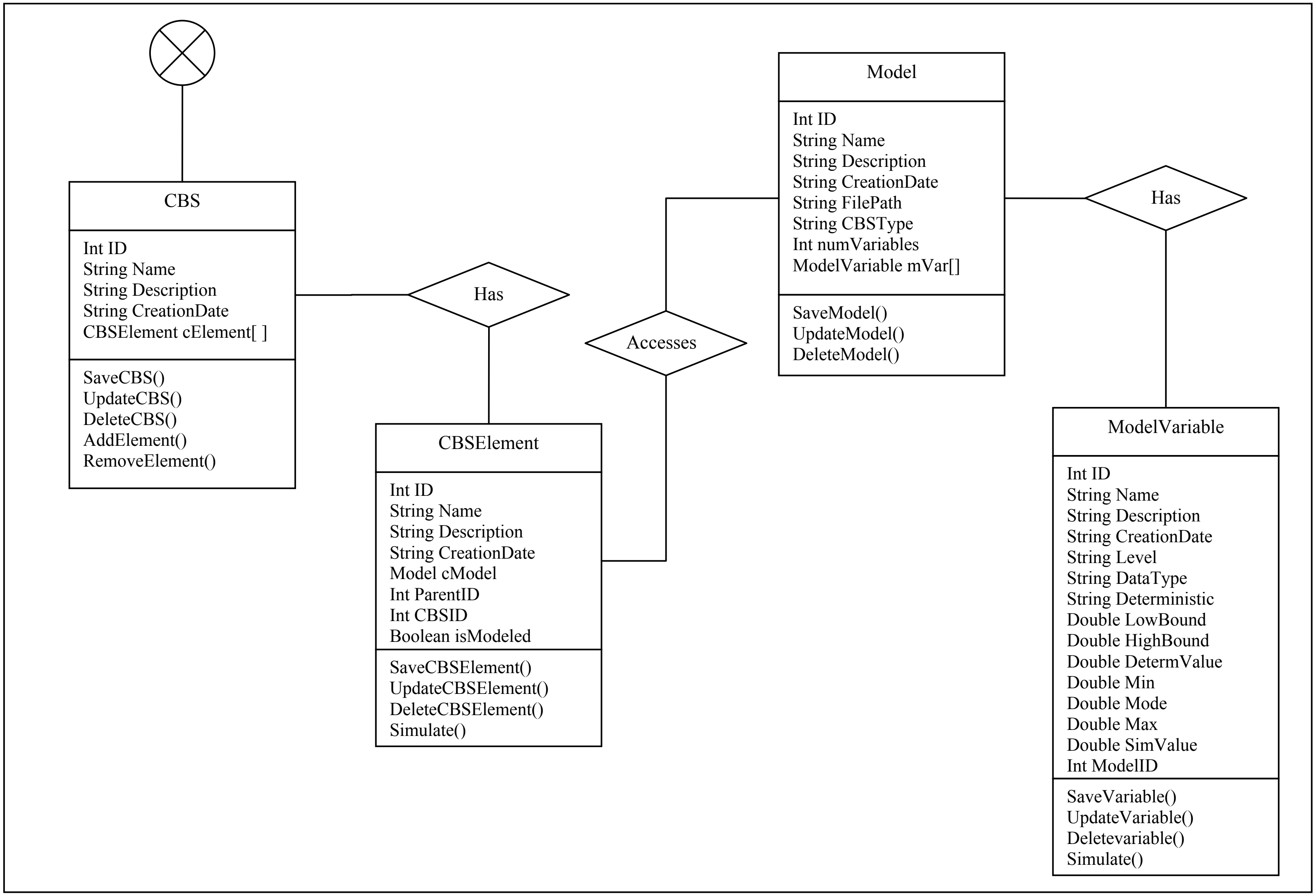


Figure 1-11: Entity Relationship Diagram for CRT-3 (cont.)

Table 1-1: Features list of CRT versions.

Capability/Feature	**CRT, version 1**	**CRT, version 2**	**CRT, version 3**
Completion/Release date	December 2001	May 2002	nearly complete at termination
Development environment	*Visual Basic 6*	*VB.net*	*VB.net*
Object orientation	minimal	class structure	fully; compete redesign (see Entity Relationship diagram in Figure 1-11)
Basis	cost process definitions	cost process definitions, defined use cases (see Appendix 1-B)	cost process definitions, enhanced use cases (see Appendix 1-C)
Underlying concepts	estimate instance, model centric	estimate instance, model centric, MEMS array-based architecture	estimate instance, model centric, MEMS array-based architecture
CBS (Cost Breakdown Structure)	hierarchical, static (imbedded in PBS)	hierarchical, dynamic/flexible (independent of, but linked to, PBS but cannot vary by PBS element), Explorer-type user interface	hierarchical, dynamic/flexible (independent of, but linked to, PBS and can vary by PBS element), enhanced Explorer-type user interface
PBS (Product Breakdown Structure)	hierarchical, dynamic/flexible w/imbedded PBS	hierarchical, dynamic/flexible (independent of, but linked to, CBS), Explorer-type user interface	hierarchical, dynamic/flexible (independent of, but linked to, CBS), enhanced Explorer-type user interface
Monte Carlo simulation engine	user selects points of uncertainty; provides roll ups, multiple measures, multiple assessment graphs; triangular distribution only; proven random variate generator	user selects points of uncertainty; provides roll ups, multiple measures, multiple assessment graphs; triangular distribution only; proven random variate generator; variance reduction technique	user selects points of uncertainty; provides roll ups, multiple measures, multiple assessment graphs; triangular distribution only; proven random variate generator; variance reduction technique; can be multithreaded
Model selection	look & feel only, internal	based on data dictionary, external, Excel only	based on data dictionary, external, Excel only
Model registration	none	using data dictionary	using data dictionary
Models per instance	one primary, fixed/imbedded	one primary, variable/flexible	one primary, multiple adjustment, variable/flexible
Data management	relational	flat file	relational, “close” to server based
Industry review	internal	industry	internal

The remainder of the section examines the modules of the system. Screen shots from the prototype illustrate how the different modules of CRT are implemented and used in typical sessions.

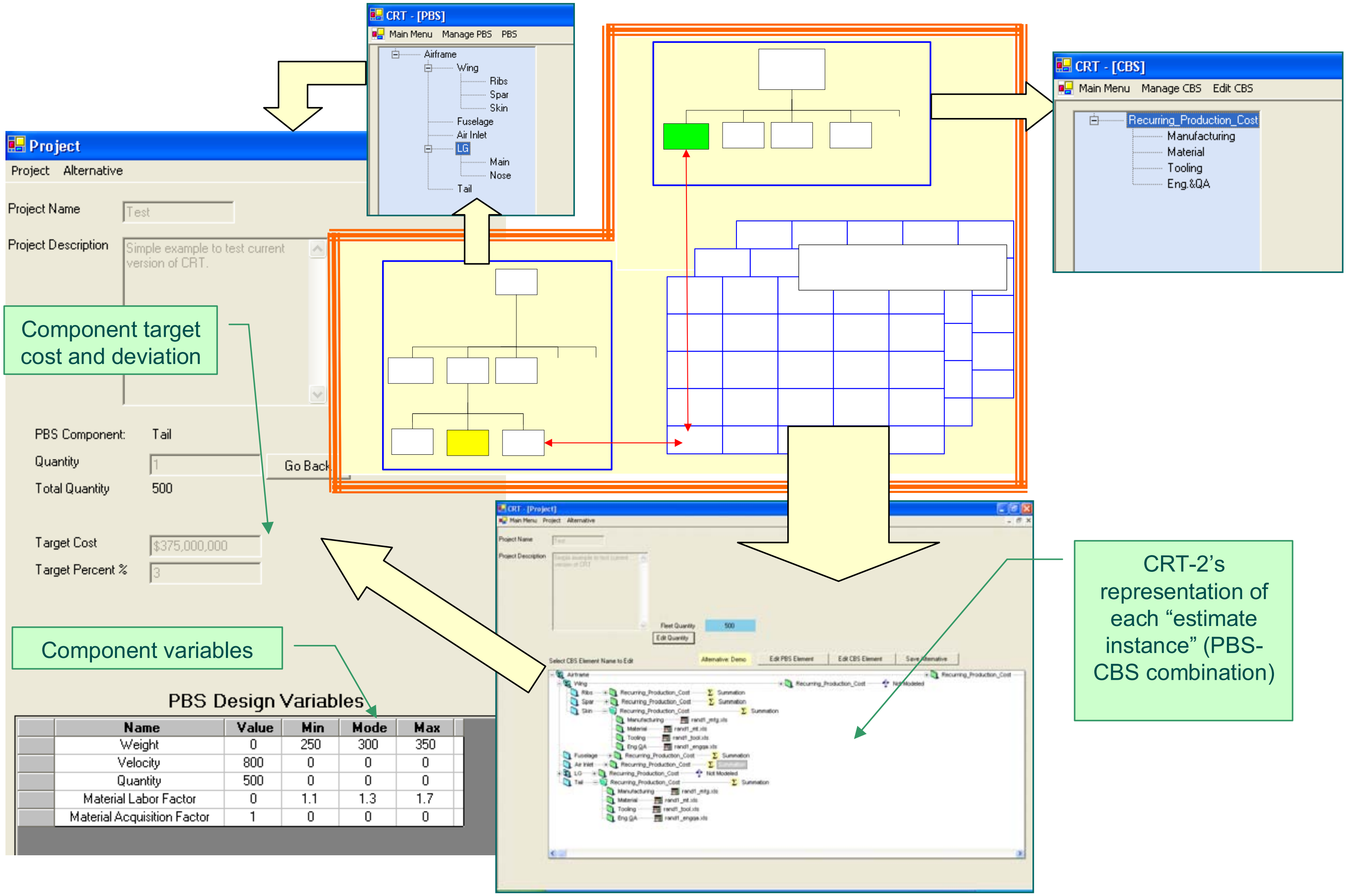
While CRT is a completely new system, it has roots, conceptually at least, in earlier systems that have been co-developed by the PI. These earlier systems are described in:

> Greenwood, A. G., "Cost Evaluation/In situ Design Cost Trades," *Anteon Corporation* and *Air Force Research Laboratory*, Contract F33615-96-D-5608, Delivery Order 34, September 2000
>
> Greenwood, A. G., and Guo, S. R., "Integrating Cost Assessment into Product/Process Design: An Object-Based Approach, *Journal of Engineering Valuation and Cost Analysis*, 1999, in review.
>
> Rais-Rohani, M., and Greenwood, A. G. "Product and Process Coupling in Multidisciplinary Design of Flight Vehicle Structures," *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 1998, St. Louis, MO.
>
> Greenwood, A. G. "Development of a Prototype to Test and Demonstrate the Manufacturing-Oriented, Design-Directed Cost Estimation (MODDCE) Framework," Summer Research Extension Program, Wright Laboratory / Manufacturing Technology Directorate, Manufacturing and Engineering Systems Division, February 1998, *Air Force Office of Scientific Research*, Contract No. AFOSR 97-0815.
>
> Greenwood, A. G. "An Approach to Enhance Cost Estimation During Product/Process Design," *Proceedings: Decision Sciences Institute Conference,* November 1997, San Diego, CA.

1.6.4 CRT Operations and Interfaces

This section provides an overview of how the CRT is used. Most of the following figures provide screen shots of the implementation within CRT of the modules that comprise the overall architecture, as was defined in Figure 1-9.

The first step in using CRT is to define or import the product breakdown structure (PBS), which contains a hierarchical definition of the system's components, and the cost breakdown structure (CBS), which contains a hierarchical definition of the cost elements to be considered in the analysis. CRT automatically creates the MEMS (Model/Estimate Management Structure) based on the PBS and CBS. This is illustrated in Figure 1-12. The interface for defining the hierarchical PBS and CBS are shown at the top of Figure 1-12. The MEMS is implemented in the CRT prototype through the "Explorer-like" tree structure shown in the screen shot in the lower center of Figure 1-12. Each PBS component contains its own CBS representation. A more detailed view of this screen is provided in Figure 1-13. As is shown in the screenshot, each PBS/CBS combination (estimate instance) is either linked to a model (discussed later), is a roll-up or summation, or is estimated by a higher-level element.



Project Name: Test

Project Description: Simple example to test current version of CRT.

PBS Component: Tail

Quantity: 1

Total Quantity: 500

Target Cost: $375,000,000

Target Percent %: 3

Fleet Quantity: 500

Component target cost and deviation

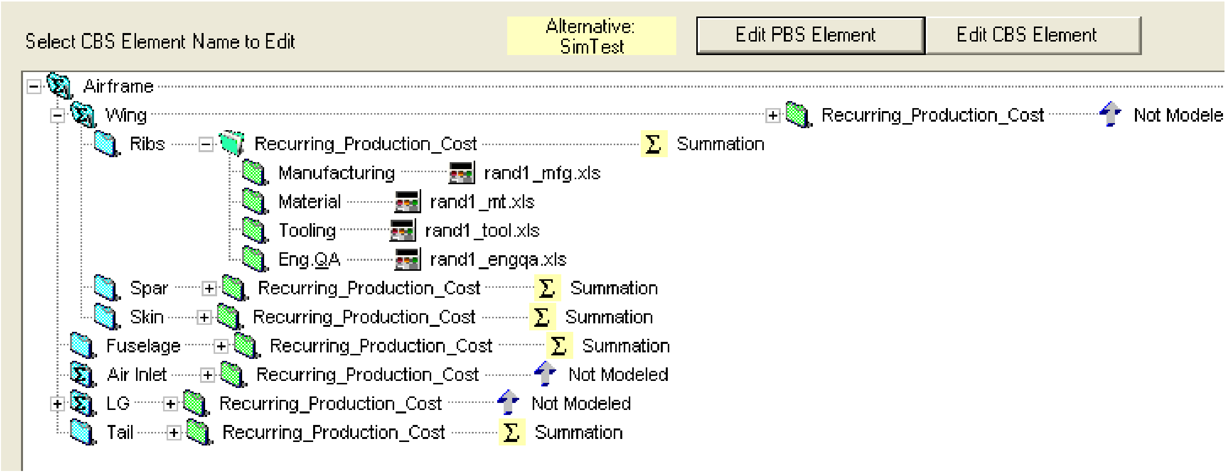
Component variables

PBS Design Variables

Name	Value	Min	Mode	Max
Weight	0	250	300	350
Velocity	800	0	0	0
Quantity	500	0	0	0
Material Labor Factor	0	1.1	1.3	1.7
Material Acquisition Factor	1	0	0	0

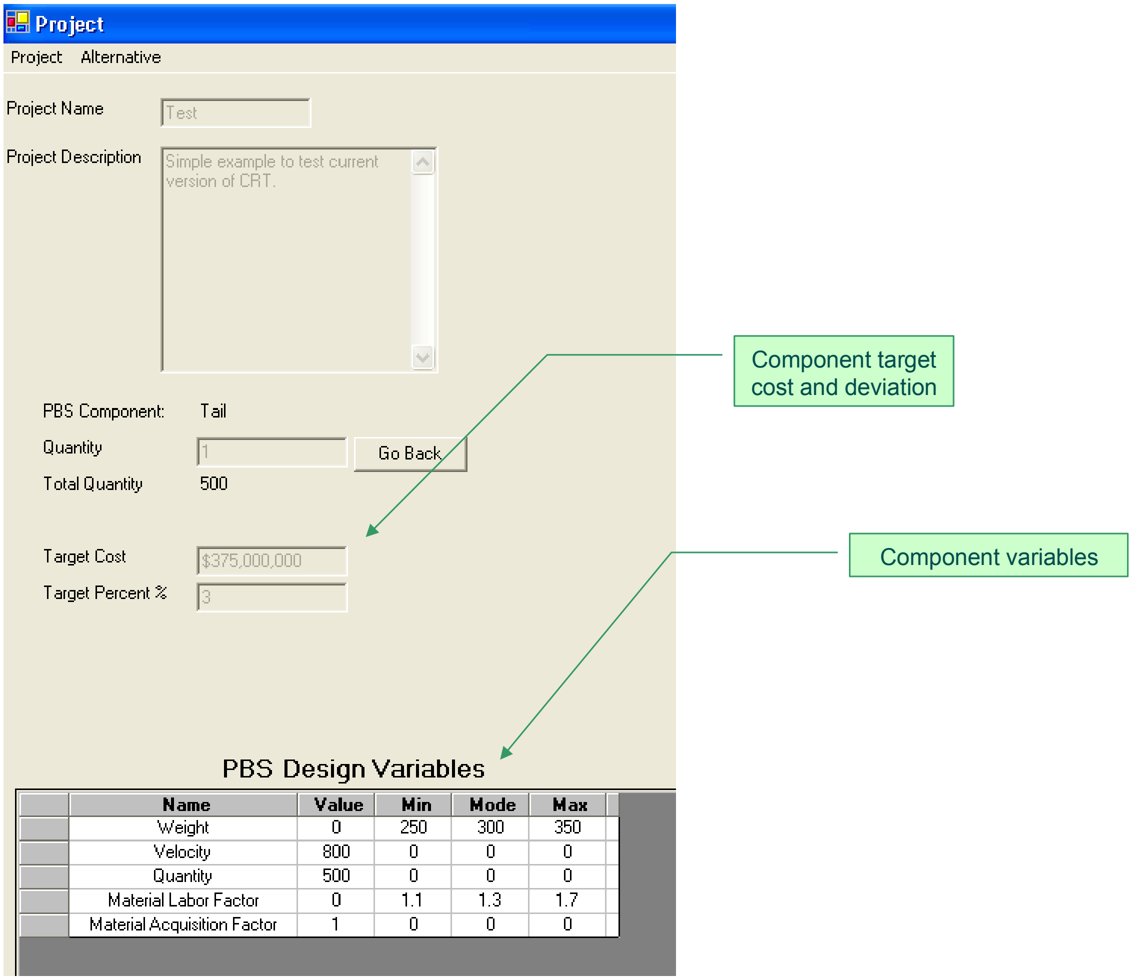
CRT-2's representation of each "estimate instance" (PBS-CBS combination)

Figure 1-12: Definition of the two hierarchical structures, the PBS and the CBS; creation of MEMS.



```
Airframe
  Wing                                                    Recurring_Production_Cost   Not Modele
    Ribs    Recurring_Production_Cost   Summation
              Manufacturing   rand1_mfg.xls
              Material   rand1_mt.xls
              Tooling   rand1_tool.xls
              Eng.QA   rand1_engqa.xls
    Spar    Recurring_Production_Cost   Summation
    Skin    Recurring_Production_Cost   Summation
  Fuselage    Recurring_Production_Cost   Summation
  Air Inlet    Recurring_Production_Cost   Not Modeled
  LG    Recurring_Production_Cost   Not Modeled
  Tail    Recurring_Production_Cost   Summation
```

Figure 1-13: Implementation of “estimate instances” in CRT; “Explorer” view of MEMS.



Project Name: Test

Project Description: Simple example to test current version of CRT.

PBS Component: Tail

Quantity: 1

Total Quantity: 500

Target Cost: $375,000,000

Target Percent %: 3

PBS Design Variables

Name	Value	Min	Mode	Max
Weight	0	250	300	350
Velocity	800	0	0	0
Quantity	500	0	0	0
Material Labor Factor	0	1.1	1.3	1.7
Material Acquisition Factor	1	0	0	0

Figure 1-14: Basic information on product’s components.

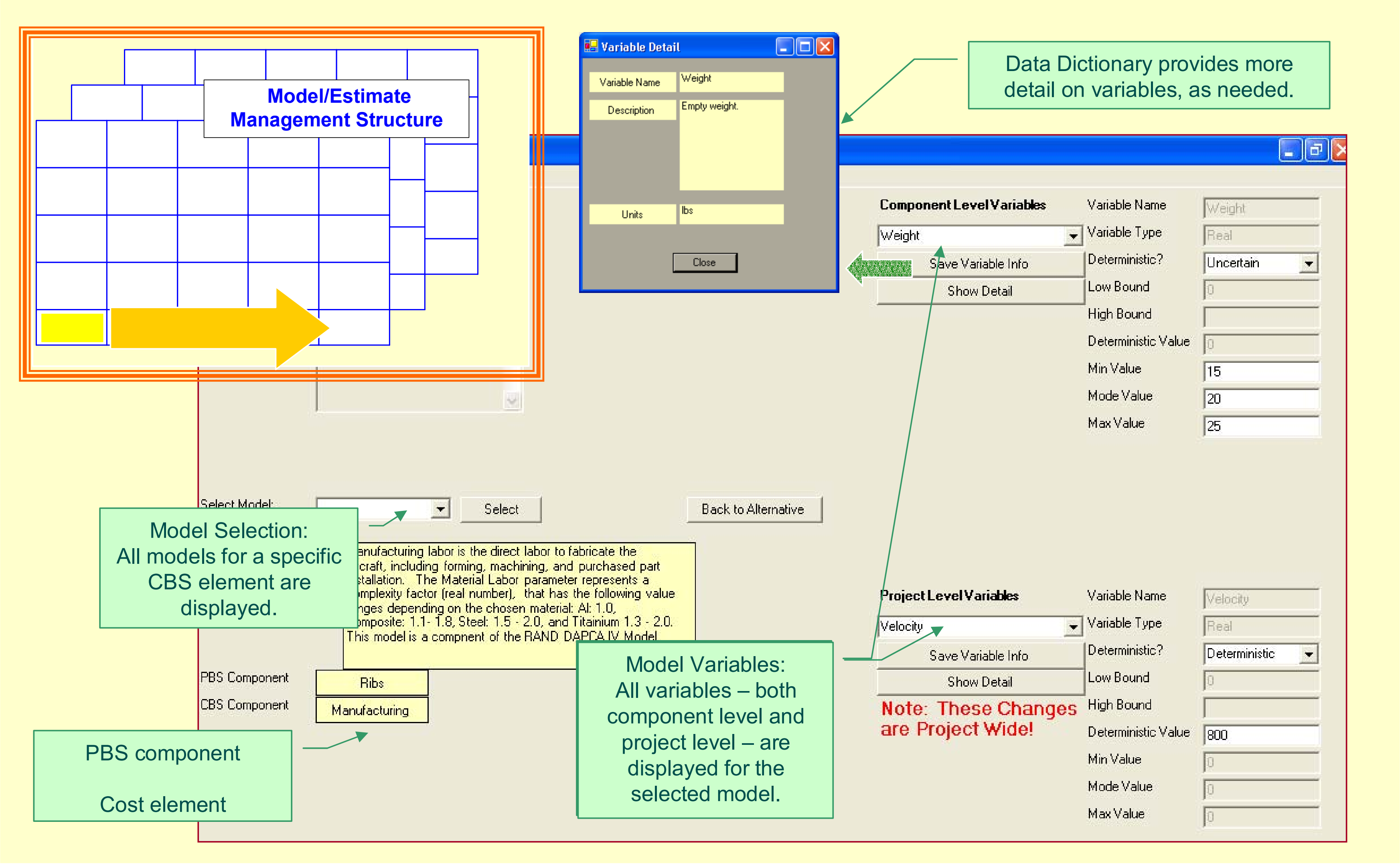
Information about each component in the PBS is shown on the interface in the left-hand portion of Figure 1-12; a closeup of the screen is provided in Figure 1-14. This screen includes the user-specified target cost and tolerable range (expressed as a percentage of the target cost) and a data table of component variable values. Once the calculation option for the alternative is invoked, the cost estimate for each component is available by clicking on a PBS node in the tree and the estimate is displayed on this screen.

As indicated earlier, models are the means to transform design variables and component characteristics into cost estimates. CRT is model centric, in that many of capabilities deal with model management. Figure 1-15 illustrates some of the model management operations. As shown in the upper left portion of Figure 1-15, model control is at the estimate instance level as defined in the MEMS.

As shown in the lower left portion of Figure 1-15, a model is selected and used to estimate the cost associated with each cost element defined in the CBS for each PBS component. In this example, ribs in an aircraft's wing are the PBS component and manufacturing cost is the CBS element. As a result, all manufacturing cost models are made available for selection through the drop-down box in the left-hand portion of the screen. Once the model has been selected, the variables used in that model – both component-level and project-level – are listed in the drop-down boxes in the right-hand portion of Figure 1-15.

This same interface is the input screen that is used to specify the value of each of the variables for each component. The inputs may be either deterministic or stochastic; if stochastic, the user needs to enter the minimum, most likely, and maximum possible values for the variable. As shown in the example in Figure 1-15, the selected model requires component weight and velocity. Weight is a component variable and its value is uncertain' therefore, the minimum, most likely, and maximum weights are entered as 15, 20, and 25, respectively. Note, if a variable is used in another cost element for the component, its value is only entered once. Velocity is a project-level and its deterministic value of 800 that is shown in Figure 1-15 may have been entered at some other time. Since it is a project-level variable, if this variable is changed, it will most likely impact other components; i.e., if another component uses a model that contains the velocity variable. If the user needs more information on any variable, e.g., a definition or its units of measure, the "Show Detail" button displays that information from the Data Dictionary.

The data dictionary utilized above (through a popup item to help the user provide values for each variable that is be used to generate a cost estimate) is an important element of the CRT. It provides variable integrity in that when models are registered into CRT (described below) either their unique variables are registered or the model is linked to variables already registered in the data dictionary. As shown in the data dictionary interface screen in Figure 1-16, each variable is defined in terms of a description, its units of measure, and synonyms. Parameters and CBS elements are also defined in the data dictionary. In CRT-3 the data dictionary will be hierarchical, i.e. the entries will grouped or categorized, to facilitate its use and maintenance.



Model/Estimate Management Structure

Data Dictionary provides more detail on variables, as needed.

Variable Detail

Variable Name: Weight

Description: Empty weight.

Units: lbs

Component Level Variables

Weight

Variable Name: Weight

Variable Type: Real

Deterministic?: Uncertain

Low Bound: 0

High Bound:

Deterministic Value: 0

Min Value: 15

Mode Value: 20

Max Value: 25

Select Model:

Model Selection: All models for a specific CBS element are displayed.

anufacturing labor is the direct labor to fabricate the craft, including forming, machining, and purchased part stallation. The Material Labor parameter represents a mplexity factor (real number), that has the following value nges depending on the chosen material: Al: 1.0, mposite: 1.1- 1.8, Steel: 1.5 - 2.0, and Titainium 1.3 - 2.0. This model is a compnent of the RAND DAPCA IV Model

PBS Component: Ribs

CBS Component: Manufacturing

PBS component

Cost element

Model Variables: All variables – both component level and project level – are displayed for the selected model.

Project Level Variables

Velocity

Note: These Changes are Project Wide!

Variable Name: Velocity

Variable Type: Real

Deterministic?: Deterministic

Low Bound: 0

High Bound:

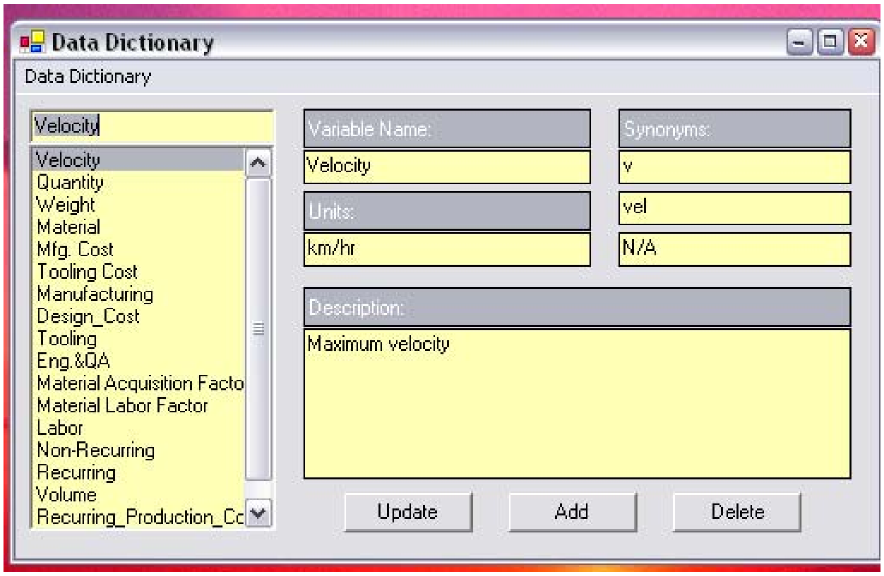
Deterministic Value: 800

Min Value: 0

Mode Value: 0

Max Value: 0

Figure 1-15: Model section and variable definition operations.



Velocity

Velocity
Quantity
Weight
Material
Mfg. Cost
Tooling Cost
Manufacturing
Design_Cost
Tooling
Eng.&QA
Material Acquisition Facto
Material Labor Factor
Labor
Non-Recurring
Recurring
Volume
Recurring_Production_Cc

Variable Name: Velocity

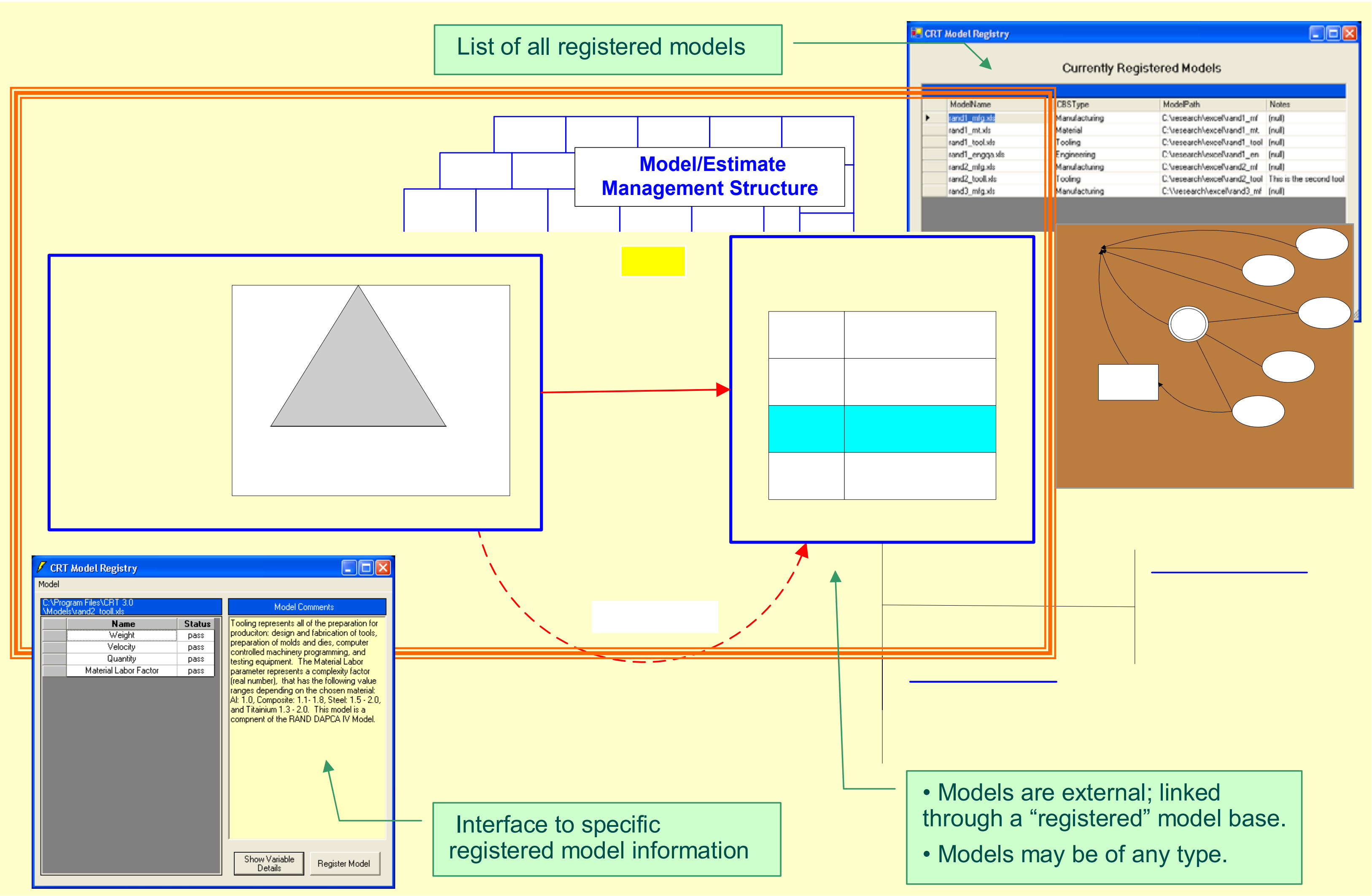
Synonyms: v
vel
N/A

Units: km/hr

Description: Maximum velocity

Figure 1-16: Data dictionary interface screen.

The operations surrounding the model base are shown in Figure 1-17. The model base contains information on all of the cost models that have been "registered" with the system. Registering requires noting which CBS element the model provides an estimate for and the variables it needs in order to perform the estimate; these need to correspond to an entry in the data dictionary. In essence, only the inputs to and output from a model are specified; internal aspects (the methodologies of the model) remain hidden to the system. The models may be of any type – e.g. parametric, analogous, detail. Presently only Excel models on the client are included; however, it is planned that future versions of the tool will link to other domains, e.g. ICE, SEER, proprietary, etc. The screen shot in the upper right corner of Figure 1-17 provides a list of all registered models and their location; the screen shot in the lower left corner of Figure 1-17 shows the information screen on a single registered model.



List of all registered models

Currently Registered Models

	ModelName	CBSType	ModelPath	Notes
▶	rand1_mfg.xls	Manufacturing	C:\research\excel\rand1_mf	(null)
	rand1_mt.xls	Material	C:\research\excel\rand1_mt.	(null)
	rand1_tool.xls	Tooling	C:\research\excel\rand1_tool	(null)
	rand1_engqa.xls	Engineering	C:\research\excel\rand1_en	(null)
	rand2_mfg.xls	Manufacturing	C:\research\excel\rand2_mf	(null)
	rand2_tooll.xls	Tooling	C:\research\excel\rand2_tool	This is the second tool
	rand3_mfg.xls	Manufacturing	C:\\research\excel\rand3_mf	(null)

C:\Program Files\CRT 3.0 \Models\rand2_tooll.xls

Name	Status
Weight	pass
Velocity	pass
Quantity	pass
Material Labor Factor	pass

Model Comments

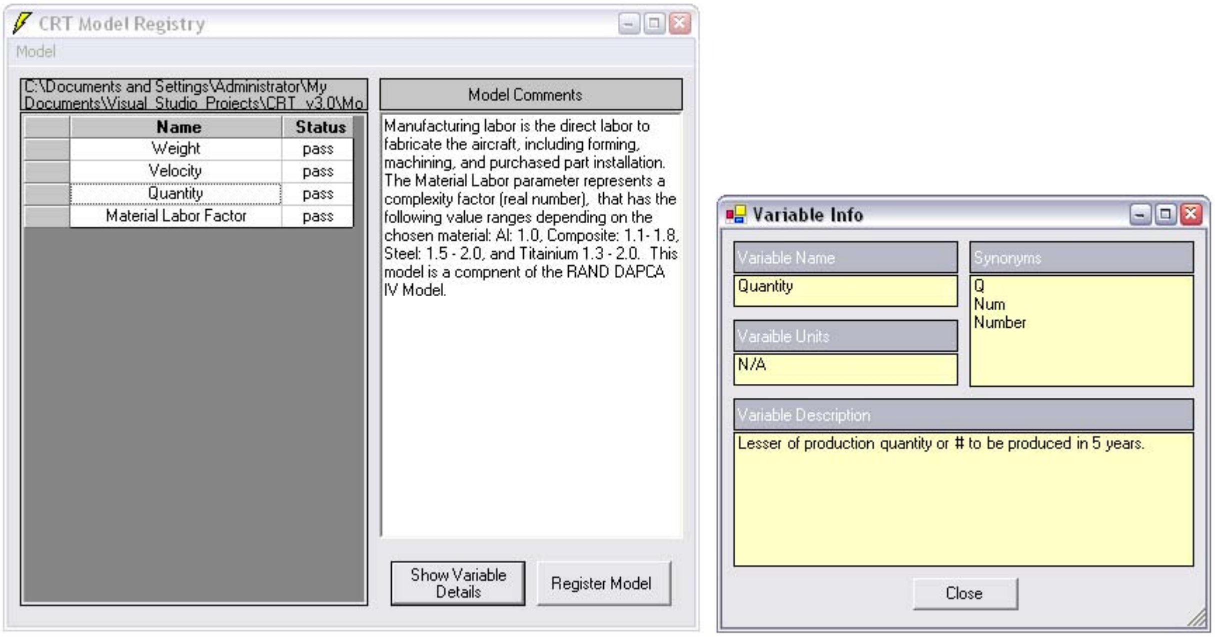
Tooling represents all of the preparation for produciton: design and fabrication of tools, preparation of molds and dies, computer controlled machinery programming, and testing equipment. The Material Labor parameter represents a complexity factor (real number), that has the following value ranges depending on the chosen material: Al: 1.0, Composite: 1.1- 1.8, Steel: 1.5 - 2.0, and Titainium 1.3 - 2.0. This model is a compnent of the RAND DAPCA IV Model.

Interface to specific registered model information

- Models are external; linked through a "registered" model base.
- Models may be of any type.

Figure 1-17: Model registration and management.

The data dictionary is also used to identify the appropriate models that can be used to estimate the cost for a CBS element and to manage the variables/parameters used in the models. Figure 1-18 shows the data dictionary in conjunction with model information screen.



C:\Documents and Settings\Administrator\My Documents\Visual Studio Projects\CRT v3.0\Mo

Name	Status
Weight	pass
Velocity	pass
Quantity	pass
Material Labor Factor	pass

Manufacturing labor is the direct labor to fabricate the aircraft, including forming, machining, and purchased part installation. The Material Labor parameter represents a complexity factor (real number), that has the following value ranges depending on the chosen material: Al: 1.0, Composite: 1.1- 1.8, Steel: 1.5 - 2.0, and Titainium 1.3 - 2.0. This model is a compnent of the RAND DAPCA IV Model.

Variable Name: Quantity

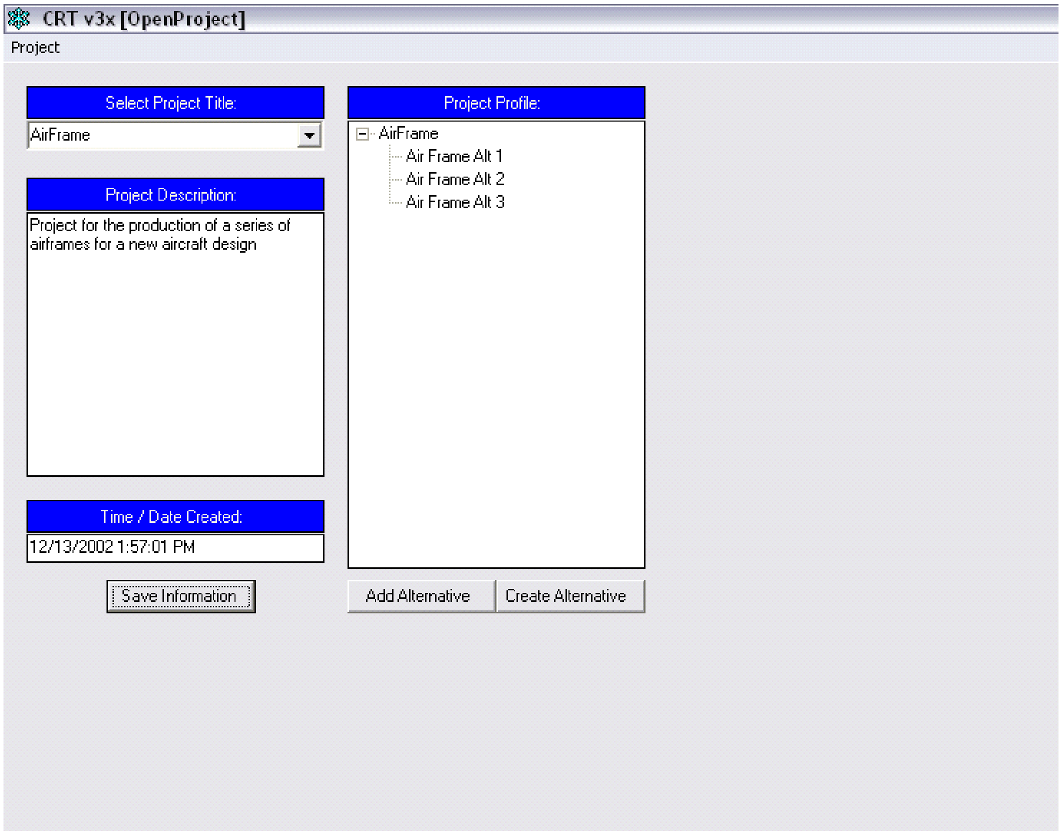
Varaible Units: N/A

Synonyms:
Q
Num
Number

Variable Description: Lesser of production quantity or # to be produced in 5 years.

Figure 1-18: Model information in registry linked to data dictionary.

Before proceeding to a discussion of the output of the CRT, we provide screen shots of a new interface that is planned for the next release, CRT-3. Figure 1-19 shows the high-level interface screen where a specific project is selected as well as the alternative that is to be considered for that project.



Select Project Title: AirFrame

Project Description: Project for the production of a series of airframes for a new aircraft design

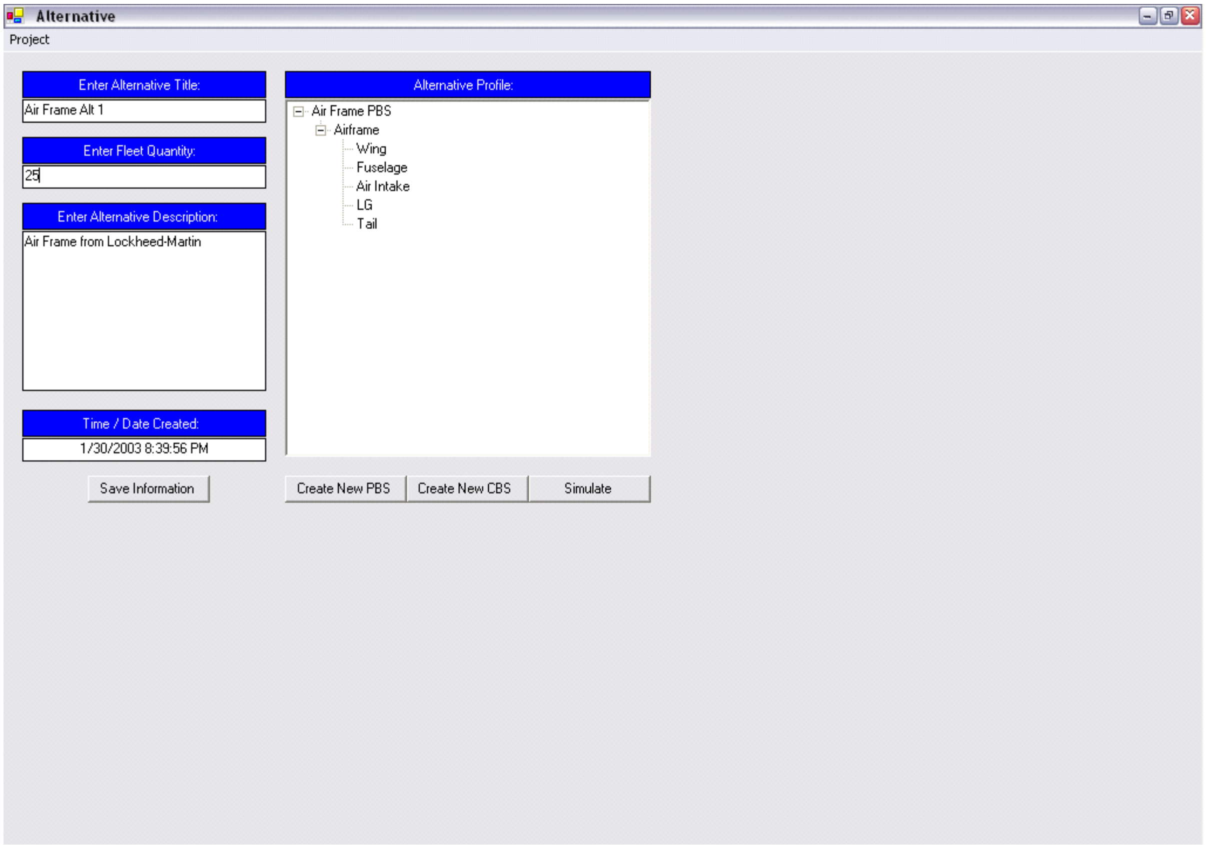
Time / Date Created: 12/13/2002 1:57:01 PM

Project Profile:

- AirFrame
  - Air Frame Alt 1
  - Air Frame Alt 2
  - Air Frame Alt 3

Figure 1-19: New interface at the projects, or highest, level.

Figure 1-20 shows the interface once a project alternative is selected. The user can either create a new PBS or CBS or simulate the current instance.



Enter Alternative Title: Air Frame Alt 1

Enter Fleet Quantity: 25

Enter Alternative Description: Air Frame from Lockheed-Martin

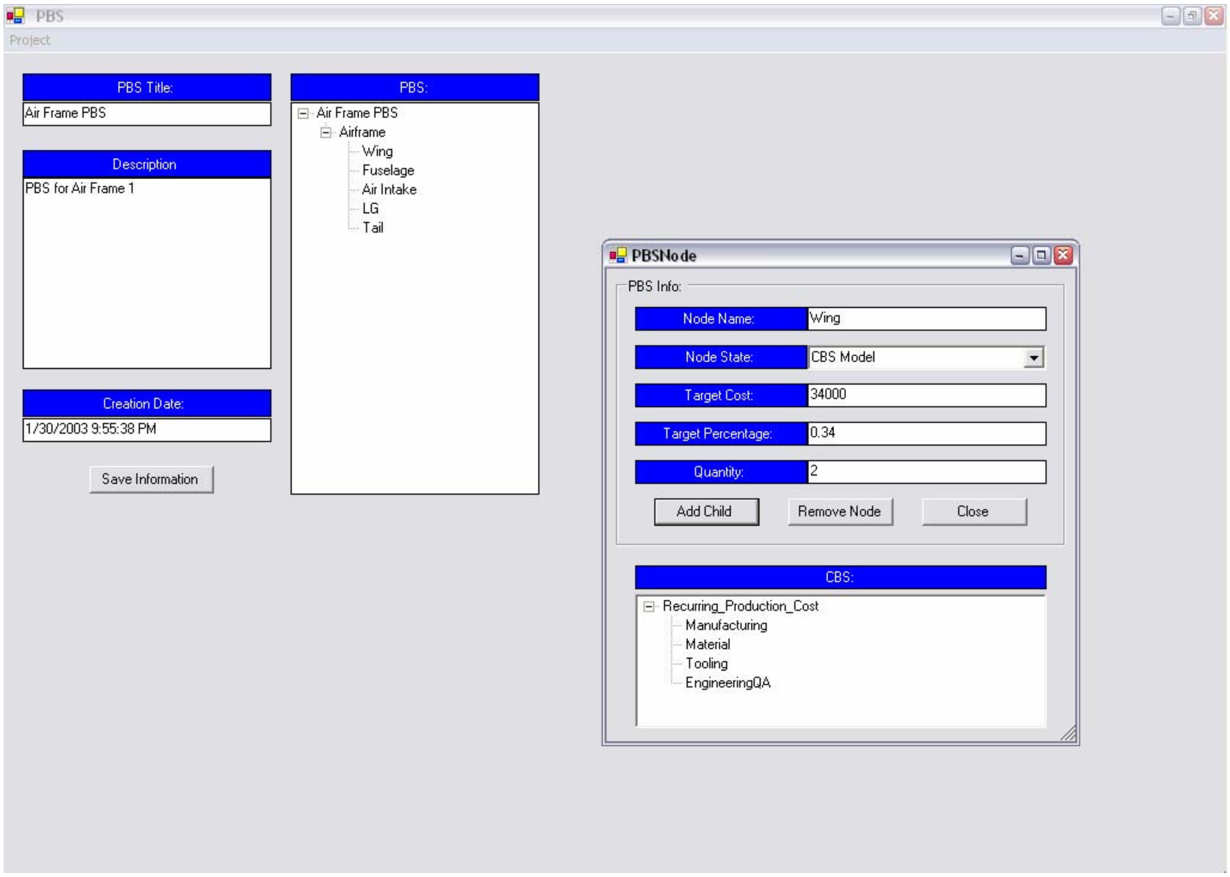
Time / Date Created: 1/30/2003 8:39:56 PM

Alternative Profile:

- Air Frame PBS
  - Airframe
    - Wing
    - Fuselage
    - Air Intake
    - LG
    - Tail

Figure 1-20: New interface for a specific project at the "alternative" level.

The popup screen shown in Figure 1-21 provides information on each PBS component, including target cost information. The hierarchy can be expanded or trimmed through this screen; i.e., components can be added or deleted. The CBS for that component is also displayed within the screen; remember that while all CBSs for an alternative are derived from the same meta CBS or template, the level of detail can vary by PBS component.



PBS Title: Air Frame PBS

Description: PBS for Air Frame 1

Creation Date: 1/30/2003 9:55:38 PM

PBS:

- Air Frame PBS
  - Airframe
    - Wing
    - Fuselage
    - Air Intake
    - LG
    - Tail

Node Name: Wing

Node State: CBS Model

Target Cost: 34000

Target Percentage: 0.34

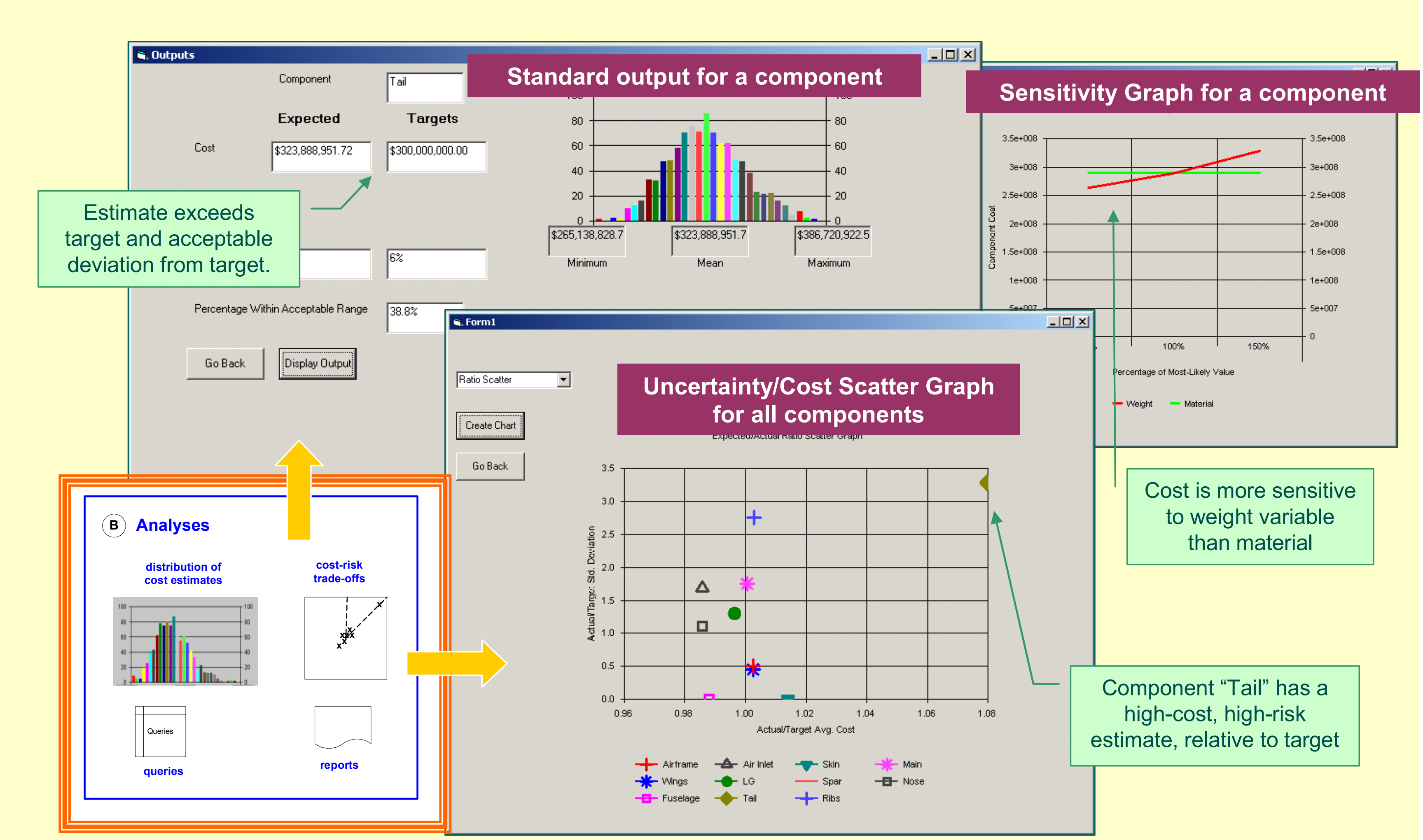
Quantity: 2

CBS:

- Recurring_Production_Cost
  - Manufacturing
  - Material
  - Tooling
  - EngineeringQA

Figure 1`-21: New interface for the PBS/CBS interface.

A sample of the various output that is available from the CRT is shown in Figure 1-22. The Standard Output screen for a component, shown in the top center of Figure 1-22, provides a histogram of the cost estimates (if there are any stochastic variables) and comparisons between expected (mean from simulation) and target values. A Sensitivity Graph or spider plot, shown in the right portion of Figure 1-22, is used at the component level to determine how much impact each variable in a cost model has on the cost estimated by the model. The Uncertainty/Risk Scatter Graph, shown in the center of Figure 1-22, considers all components in the system. It is used to identify those components that have higher estimated cost and/or higher estimated risk relative to the targeted values. Each of these output screens are shown individually in Section 3, Briefing Charts.



Standard output for a component

Outputs

Component: Tail

	Expected	Targets
Cost	$323,888,951.72	$300,000,000.00
		6%

Percentage Within Acceptable Range: 38.8%

Minimum	Mean	Maximum
$265,138,828.7	$323,888,951.7	$386,720,922.5

Estimate exceeds target and acceptable deviation from target.

Sensitivity Graph for a component

Cost is more sensitive to weight variable than material

Uncertainty/Cost Scatter Graph for all components

Form1

Ratio Scatter

Component “Tail” has a high-cost, high-risk estimate, relative to target

Figure 1-22: Output screens from the CRT

1.6.5 Output / Performance Measures

Since some of the model inputs are random variables, the component costs and system cost are random variables. Therefore, the estimated mean component cost and total system cost are based on *R* simulation replications and are expressed in by the first two equations below. Similarly, the standard deviations of component cost and total system cost that is obtained from the simulation are expressed in the next two equations.

$$E[C_j] = \frac{1}{R}\sum_{i=1}^{R} C_{ij}$$

$$E[Cost_{Sys}] = \frac{1}{R}\sum_{i=1}^{R}\sum_{j=1}^{n} C_{ij}$$

$$S[C_j] = \sqrt{\frac{1}{R-1}\sum_{i=1}^{R}\left(C_{ij} - E[C_j]\right)^2}$$

$$S[Cost_{Sys}] = \sqrt{\frac{1}{R-1}\sum_{i=1}^{R}\left(Cost_{Sys_i} - E[Cost_{Sys}]\right)^2}$$

As described above and as represented in the equation below, the user provides for each component *j* in the PBS both a target cost, C_j^T, and an acceptable percentage deviation from the target cost, $DevC_j^T$. This results in an acceptable target cost interval for each component:

$$CI_j^T = \left\{C_j^T - C_j^T \times DevC_j^T\ ,\ C_j^T + C_j^T \times DevC_j^T\ \right\}$$

It is assumed that the range of this interval represents six standard deviations (i.e., +/- $3\sigma_j^T$) from the target cost. Therefore, the target standard deviation is:

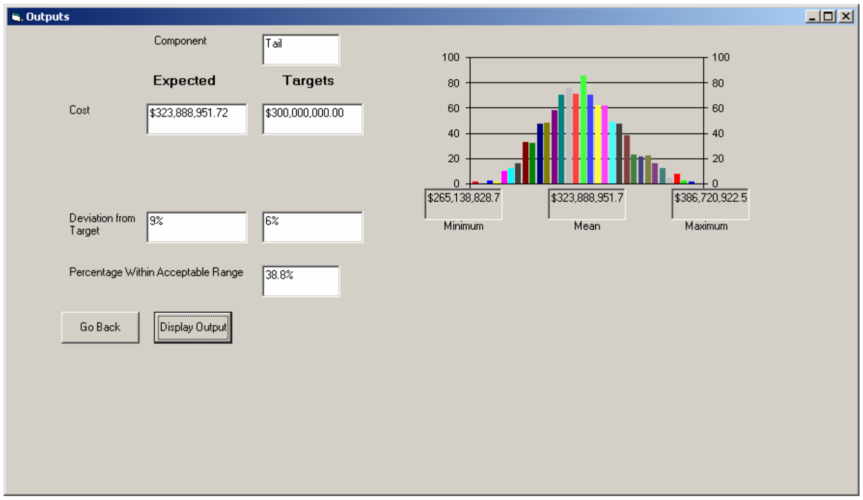
$$\sigma_j^T = \left\{\left(C_j^T + C_j^T \times DevC_j^T\right) - \left(C_j^T - C_j^T \times DevC_j^T\right)\right\}/6 = {}^1\!/_3\, C_j^T \times DevC_j^T$$

The simulation estimates the mean percentage deviation from the target cost by the following equation.

$$E\left[DevC_j^T\right] = \frac{\sum_{i=1}^{R}\left|C_{ij} - C_j^T\right|}{R \times C_j^T} \times 100$$

As a measure of uncertainty/risk, the CRT also calculates the percentage of simulation runs where the estimated component cost C_{ij} is within the acceptable target cost interval CI_j^T.

The results of the simulation are displayed in several formats, as shown in Figure 22, to facilitate analysis. An example of component-level output is provided in Figure 23. It includes a comparison of expected cost from the simulation ($E[C_j]$) and its specified target cost (C_j^T), a comparison of the expected percentage deviation from the target ($E\left[DevC_j^T\right]$) and the specified acceptable deviation ($DevC_j^T$), and the percentage of cost estimates that fell within the acceptable target interval (CI_j^T). A histogram of the costs from each replication is also provided, as shown in the right-hand portion of Figure 23.



Outputs

Component: Tail

	Expected	Targets
Cost	$323,888,951.72	$300,000,000.00
Deviation from Target	9%	6%

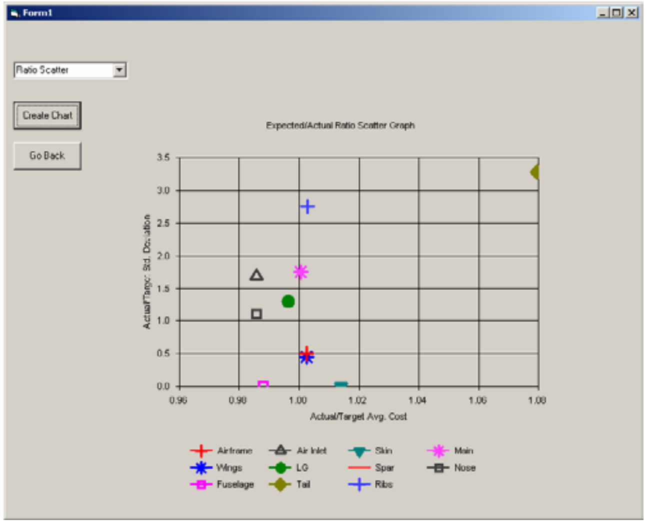
Percentage Within Acceptable Range: 38.8%

Go Back | Display Output

Minimum	Mean	Maximum
$265,138,828.7	$323,888,951.7	$386,720,922.5

Figure 1-23: Example component-level simulation output from the CRT.

The CRT also generates an uncertainty/cost scatter graph that is used to identify which components exhibit high/low cost and high/low uncertainty. An example graph is shown in Figure 24. Each component is plotted based on two ratios. The cost ratio, represented by the X-axis, is the estimated expected cost, $E[C_j]$, divided by the specified target cost for the component C_j^T. The risk ratio, represented by the Y-axis, is the expected deviation from the target cost, $E\left[DevC_j^T\right]$, divided by the acceptable deviation for the component, $DevC_j^T$. Notice the landing gear component (denoted LG and represented by a circle in Figure 24) is close to being on target with respect to both risk and cost (both ratios ≈ 1). Also, the ribs component, represented by a + in Figure 24, is near its target cost but has high risk; the tail component, represented by a diamond, exhibits both high cost and high risk. CRT also performs sensitivity analyses; output is not shown due to space considerations.



Form1

Ratio Scatter

Create Chart | Go Back

Figure 1-24: Example uncertainty/cost scatter graph from the CRT.

1.6.6 Future direction for the CRT

The following are some of our ideas for further development of the CRT to evolve to a comprehensive IDCT Tool. Other activities and functionality should be identified through interaction with industry. Also, the features that are incorporated should be prioritized based on industry feedback.

- Permit “partitioned” risk analyses, i.e. perform simulations only on selected sections of the PBS.
- Exploit multi-threading capability of the operating system, and investigate parallel processing, to speed up the simulation analyses.
- Migrate to a client/server environment.
- Investigate the application of XML and web portals as enabling technologies.
- Link to CAD/CAE system in order to facilitate entering values for the component-level design variables in the PBS.
- Link to other cost models/systems.
- Enhance reporting capability and database queries.
- Develop hierarchical data dictionary in order to facilitate use and searches.
- Support other types of probability distributions to represent uncertainty in input variables.
- Provide capability to “fit” the output to theoretical distributions; i.e., better characterize cost distributions.
- Incorporate the application of “adjustment” models.
- Test, evaluate, and deploy in industry and academe.

1.7 Project Team

The following tables provide a list of the project participants including faculty, students, and industry reviewers.

1.7.1 Faculty

Name	Role	Contribution	Funded
Dr. Allen G. Greenwood, PE Department of Industrial Engineering Mississippi State University	Principal Investigator	System architect and conceptual designer Life cycle cost process, modeling, and analysis Risk analysis and simulation Student and Thesis advisor Co-author on 1 journal article, 4 conference papers published, 2 in preparation Project manager; Industry Liaison	Yes
Dr. Masoud Rais-Rohani, PE Department of Aerospace Engineering Mississippi State University	Researcher	Advisor on design of aerospace structures Multidisciplinary Design Optimization	Yes

1.7.2 Student Participants

Name (all from MSU)	Level	Contribution	Funded
Stephen W. Ormon	M.S.	Thesis: "Development of a Hierarchical Model-Based Design Decision-Support Tool for Assessing Uncertainty of Cost Estimates," 2002. Cost Estimating process Risk analysis and simulation Co-author 1 Journal article Co-author 4 Conference proceedings Developer of CRT-1, CRT-2, and RPM	Yes
Judy Liaw	M.S.	Thesis: "Definition and Representation of Requirements Engineering/Management: A Process-Oriented Approach," 2002. Co-author 1 Journal article (in preparation)	Yes
Shunri R. Guo	Ph.D.	Dissertation Proposal: "An Intelligent Integration Scheme for Manufacturing Cost Estimation Systems."	Yes

Travis W. Hill	M.S.	Design and development of CRT-3; object-oriented design, database design, software engineering	Yes
Praveen Gilda	M.S.	Research on Product Design Process, STEP, PDM.	Yes
Thiyagarajan Venugopalan	M.S.	Research on Simulation Based Design	Yes
Tai Chi Wu	Ph.D.	Research on Model management, XML.	No
Rita L. Endt	Ph.D.	Research on capabilities needed for life-cycle cost analyses. Investigation of MAAP and EDCAS	Yes
Mona C. Shelton, PE	Ph.D.	Research on affordability analysis, life-cycle cost	No
Marray Harris	MBA	Research on web portals Website design and development	Yes
Derrick Pratt	MBA	Website design and development	Yes
Charles Liggett	UG	WebEx and research support	Yes
Casey Dunnagan	UG	Research web-based collaborative services; WebEx	Yes

1.7.3 Industry Reviewers

Michael Bailey	General Electric Aircraft Engines
Michael Cronin	Cognition Corporation
John Fondon	Northrop Grumman (retired)
Peter Frederic	Tecolote Research, Inc.
Brian Glauser	Cognition Corporation
Joseph Jaworski	Pratt & Whitney
Mohsen Rezayat	SDRC
Steve Rogers	Acquisition Logistics Engineering
Don Shrader	TechniRep, Inc.
Ron Shroder	Frontier Technology, Inc.
Harmon Withee	Tecolote Research Inc.

1.8 Summary of Accomplishments

The following is a brief summary of the accomplishments of this project.

1. Built upon and significantly extended the cost engineering concepts and methodologies that were developed in prior work, e.g. the precursor study provided in Section 4.

2. Defined the following processes that provide the foundation for the development of the current IDCT Tool and future design/cost decision support systems:
   a. cost estimation process,
   b. requirements engineering and management processes, and
   c. product design process.

3. Developed a cost analysis decision-support system architecture – MEMS (Model/Estimate Management System) -- that:
   a. effectively couples flexible product and cost hierarchies through "cost instances"
   b. is model-centric, in that:
      i. models are linked to cost instances
      ii. related alternatives are "packaged" as projects
      iii. facilitates model selection

4. Developed 2+ versions of a working prototype for the IDCT Tool.
   a. utilizes MEMS architecture
   b. encourages the use of variable-complexity models,; i.e., models change as the design and information evolve
   c. built with an object-orientation; relational database driven (close to client server)
   d. based on cost analysis and design processes defined above
   e. incorporates model registration and model management
   f. incorporates data dictionary in model registration and model selection processes
   g. address the impact of design-variable uncertainty on component and system cost
      i. user interface for specifying component uncertainty
      ii. simulation engine to determine risk
      iii. various means of analyses to help assess risk from results of simulation
   h. reviewed by industry experts

5. Developed a prototype for a simulation-based, reliability prediction system that compliments the IDCT Tool.

6. Disseminated the work in this project through 3 journal articles (1 published, 2 in preparation) and 3 conference papers, and 2 masters theses. Additional publications are planned.

1.9 References

The following are sources specifically referenced in this section. Each of the publications that are included in Section 2 has its own set of references. Section 4, the Foundation Document, also has its own set of references, which is much more extensive since that study involved a literature review of the field.

Greenwood, A. G. "An Approach to Enhance Cost Estimation During Product/Process Design," *Proceedings: Decision Sciences Institute Conference,* November 1997, San Diego, CA.

Greenwood, A. G. “Development of a Prototype to Test and Demonstrate the Manufacturing-Oriented, Design-Directed Cost Estimation (MODDCE) Framework,” Summer Research Extension Program, Wright Laboratory / Manufacturing Technology Directorate, Manufacturing and Engineering Systems Division, February 1998, *Air Force Office of Scientific Research*, Contract No. AFOSR 97-0815

Greenwood, A. G., “Cost Evaluation/Insitu Design Cost Trades,” *Anteon Corporation* and *Air Force Research Laboratory*, Contract F33615-96-D-5608, Delivery Order 34, September 2000.

Greenwood, A. G., and Guo, S. R., “Integrating Cost Assessment into Product/Process Design: An Object-Based Approach, *Journal of Engineering Valuation and Cost Analysis*, 1999, in review

Greenwood, A. G. and Liaw, J. A. “A Framework for Engineering and Managing Requirements,” in preparation for submission to *IEEE Transactions on Engineering Management.* (Full paper is provided in Section 2.5)

Greenwood, A. G. and Ormon, S. W. “A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk,” *Proceedings of the Decision Sciences Institute Conference*, November 2002, San Diego. (Full paper is provided in Section 2.2)

Greenwood, A. G. and Ormon, S. W. “Development of a Generic Cost Estimation Process,” completed; to be submitted for review to be presented at the *ASEM National Conference,* October 2003. (Full paper is provided in Section 2.3.)

Liaw, J. “Definition and Representation of Requirements Engineering/Management: A Process-Based Approach,” Masters Thesis, Department of Industrial Engineering, Mississippi State University, May 2002

Ormon, S. W. “Development of a Hierarchical, Model-Based Design Decision-Support Tool for Assessing the Uncertainty of Cost Estimates,” Masters Thesis, Department of Industrial Engineering, Mississippi State University, May 2002.

Ormon, S. W., Cassady, C. R., and Greenwood, A. G. “Reliability Prediction Models to Support Conceptual Design,” *IEEE Transactions on Reliability*, vol. 51, no. 2, June 2002, pp. 151-157. (Full paper is provided in Section 2.1.)

Ormon, S. W., Cassady, C. R., and Greenwood, A. G. “A Simulation–Based Reliability Prediction Model for Conceptual Design,” *2001 Proceedings of the Annual Reliability and Maintainability Symposium*, January 2001, Philadelphia, pp. 433-436. Note this paper was the winner of the *The Society of Reliability Engineers’ Stan Ofsthun Best Paper Award.*

Ormon, S. W., and Greenwood, A. G. “A Comparison of Traditional and Object-Oriented Systems Analysis Tools,” *10th Annual Industrial Engineering Research Conference (IERC)*, May 2001, Dallas. (Full paper is provided in Section 2.4)

Ormon, S. W., and Greenwood, A.G. “The Use of Common Random Numbers with Monte Carlo Simulation,” Technical Note, Department of Industrial Engineering, Mississippi State University, May 2002.

Rais-Rohani, M., and Greenwood, A. G. “Product and Process Coupling in Multidisciplinary Design of Flight Vehicle Structures,” *Proceedings of the 7th AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization*, September 1998, St. Louis, MO

Raymer, D.P., *Aircraft Design: A Conceptual Approach*, 3rd Ed., American Institute of Aeronautics and Astronautics, Reston VA., 1999

Restar, S.A., Rogers, J.C., and Ronald, W.H., *Advanced Airframe Structural Materials, A Primer and Cost Estimating Methodology*, RAND Corporation Report R-4016-AF.

Venugopalan, T., and Greenwood, A.G. “Simulation-Based Design: Definition and Review,” Technical Note, Department of Industrial Engineering, Mississippi State University, May 2001.

Wu, T.C., and Greenwood, A.G. “Model Management: Definition and Review,” Technical Note, Department of Industrial Engineering, Mississippi State University, November 2002.

1.10 Section 1 APPENDIXES

A. Acronym Glossary

B. Definition of Use Cases for CRT-2, current version

C. Definition of Use Cases for CRT-3, next version

D. Cost Model Used in Example

E. Capabilities by Development Phase

F. Preliminary Capabilities List

1.10.1 Appendix 1-A: Acronym List

ASCET	Aircraft Systems Cost Estimating Tool
AFRL	Air Force Research Laboratory
AIMMS	Advanced Integrated Multidimensional Modeling System
CAD	Computer Aided Design
CER	Cost Estimating Relationship
CBS	Cost Breakdown Structure
COTS	Commercial Off The Shelf
CRT	Cost Risk Tool
DBMS	Data Base Management System
DSS	Decision Support System
ECM	Enterprise Cost Management
EDCAS	Equipment Designer's Cost Analysis System
ER	Entity Relationship
FIPER	Federated Intelligent Product EnviRonment
ICOM	Input Control Output Mechanism
IDCT	In situ Design Cost Tradeoff
IPPD	Integrated Product Process Development
LCC	Life-Cycle Cost
MAAP	Monterey Activity-based Analysis Platform
MEMS	Model/Estimate Management System
MMS	Model Management System
MS	Microsoft
MSU	Mississippi State University
M/T	Methodologies/Technologies

O&S	Operations and Support
PBS	Product Breakdown Structure
PDM	Product Data Management
PI	Principal Investigator
RPM	Reliability Prediction Models
RUP	Rational Unified Process
SBD	Simulation Based Design
STEP	STandard for the Exchange of Product model data
TAPSI	Tool for Aircraft Structures based on Process Information
VB	Visual Basic
XML	eXtensible Markup Language

1.10.2 Appendix 1-B: Definition of Use Cases for CRT-2, current version

1 Create New PBS/CBS
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create PBS / Create CBS
Purpose of Function: Allows the user to generate a product breakdown structure.
Sequence of Actions for Function:

1. The user indicates to the system that he/she wishes to create a new PBS.
2. The system prompts the user for a PBS title.
3. The user enters information.
4. The system generates a blank PBS.
5. The user indicates to the system to add nodes.
6. The system prompts user for element name.
7. The user enters element name.
8. The system adds PBS element to the PBS.
9. The user repeats process until PBS is complete.

2 Edit Existing PBS/CBS Element Names
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Edit PBS Element Names / Edit CBS Element Names
Purpose of Function: Allows the user to edit a product breakdown structure.
Sequence of Actions for Function:

1. The user opens existing PBS.
2. The system displays PBS.
3. The user selects PBS element to edit and indicates to the system to edit element.
4. The system prompts the user for a new PBS element name.
5. The user enters information.
6. The system updates the selected PBS element.

3 Create New Project
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create Project
Purpose of Function: Allows the user to create a Project.
Sequence of Actions for Function:

1. The user indicates to the system that he/she wishes to create a new Project.
2. The system prompts the user for a Project name and description.
3. The user enters information.
4. The system generates a blank Project.

4 Create New Alternative
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create Alternative
Purpose of Function: Allows the user to generate Alternative for consideration.
Sequence of Actions for Function:

1. The user opens a Project indicates to the system create a new Alternative.
2. The system prompts the user for an Alternative name.

3. The user enters Alternative name.
4. The system prompts the user for the CBS to utilize.
5. The system prompts the user for the PBS to utilize.
6. The user selects PBS.
7. The system prompts the user for fleet quantity under consideration.
8. The user enters fleet quantity.
9. The system generates blank Alternative.

5 Edit Alternative PBS Nodes

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Edit Alternative
Purpose of Function: Allows the update PBS nodes of an Alternative.
Sequence of Actions for Function:

1. The user selects node to edit and indicates to the system to edit node.
2. The system displays node summary to be edited.
3. The user updates node information (i.e. state, quantity, target cost, and target percentage).
4. The system stores updated information.

6 Edit Alternative CBS Nodes

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Edit Alternative
Purpose of Function: Allows the update CBS nodes of an Alternative.
Sequence of Actions for Function:

1. The user selects node to edit and indicates to the system to edit node.
2. The system displays node summary to be edited.
3. The user updates node information (i.e. Excel model, simulation type, deterministic value, minimum, mode, maximum, low value, high value).
4. The system stores updated information.

7 Simulate

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Simulate
Purpose of Function: Allows the user to simulate variables for analysis.
Sequence of Actions for Function:

1. The user indicates to the system to run simulation.
2. The system prompts the user for number of replications.
3. The user enters number of replications.
4. The system stores number of replication to use.
5. The user initiates simulation.
6. The system runs simulation.

8 Obtain Output

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Output
Purpose of Function: Allows the user to view output in a graphical format.
Sequence of Actions for Function:

1. The user selects output type (i.e. Sensitivity Analysis, Cost/Risk Scatter graph).
2. The system displays requested output.

1.10.3 Appendix 1-C: Definition of Use Cases for CRT-2, current version

1 Create New Project
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create Project
Purpose of Function: Allows the user to create a Project.
Sequence of Actions for Function:

1. The user indicates to the system to create a new Project.
2. The system prompts the user for a Project name and description.
3. The user enters information.
4. The system prompts the user for Alternatives.
5. User enters information.
6. The system stores the information in the database and generates the Project.

2 Create New Alternative
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create Alternative
Purpose of Function: Allows the user to generate a new Alternative for consideration.
Sequence of Actions for Function:

1. The user indicates to the system to create a new Alternative.
2. The system prompts the user for Alternative information (i.e. name, fleet quantity, PBS, CBS).
3. The user enters Alternative information.
4. The system stores the information in the database, generates an Alternative from the given information and displays the Alternative.

3 Add Alternative to Project
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create Alternative
Purpose of Function: Allows the user to add existing Alternatives to a Project.
Sequence of Actions for Function:

1. The user opens or creates a Project.
2. The system displays Project.
3. The user indicates to the system to add an Alternative.
4. The system prompts the user for Alternative.
5. The user selects Alternative.
6. The system stores the information in the database, and adds the Alternative to the Project.

4 Create New PBS
Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create PBS
Purpose of Function: Allows the user to generate a product breakdown structure.
Sequence of Actions for Function:

1. The user indicates to the system to create a new PBS.
2. The system prompts the user for a PBS title.
3. The user enters information.
4. The system generates a blank PBS.

5. The user indicates to the system to add nodes.
6. The system prompts user for element information (i.e. state, quantity, target cost, and target percentage).
7. The user enters element information.
8. The system adds PBS element to the PBS, and stores information in database.
9. The user repeats process until PBS is complete.

5 Create New CBS

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Create CBS
Purpose of Function: Allows the user to generate a cost breakdown structure.
Sequence of Actions for Function:

1. The user indicates to the system to create a new CBS.
2. The system prompts the user for a CBS title.
3. The user enters information.
4. The system generates a blank CBS.
5. The user indicates to the system to add nodes.
6. The system prompts user for element information (i.e. Excel model, simulation type, deterministic value, minimum, mode, maximum, low value, high value).
7. The user enters element information.
8. The system adds CBS element to the CBS, and stores information in database.
9. The user repeats process until CBS is complete.

6 Edit Existing PBS Elements

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Edit PBS Element
Purpose of Function: Allows the user to edit a product breakdown structure.
Sequence of Actions for Function:

1. The user opens existing PBS.
2. The system displays PBS.
3. The user selects PBS element to edit and indicates to the system to edit element.
4. The system prompts the user for new PBS element information.
5. The user updates information.
6. The system prompts the user for confirmation of changes.
7. The user enters confirmation.
8. The system updates the selected PBS element.

7 Edit Existing CBS Elements

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Edit CBS Element
Purpose of Function: Allows the user to edit a cost breakdown structure.
Sequence of Actions for Function:

1. The user opens existing CBS.
2. The system displays CBS.
3. The user selects CBS element to edit and indicates to the system to edit element.
4. The system prompts the user for new CBS element information.
5. The user updates information.
6. The system prompts the user for confirmation of changes.
7. The user enters confirmation.
8. The system updates the selected CBS element.

8 Simulate

Involved User Classes: User
Involved External Systems: NA
Mode of Operation: Normal
Function Name: Simulate
Purpose of Function: Allows the user to simulate variables for analysis.
Sequence of Actions for Function:

1. The user indicates to the system to run simulation.
2. The system prompts the user for number of replications.
3. The user enters number of replications.
4. The system runs simulation, stores output and prompts the user for output type (i.e. Sensitivity Analysis, and Cost/Risk Scatter graph) to display.
5. The user selects output type.
6. The system displays output in graphical form.

1.10.4 Appendix 1-D: Cost Model Used in Example

Although CRT is capable of managing multiple models, the example uses the following models for each cost element of the CBS. The parametric models are from an aircraft design book and a RAND report:

Raymer, D.P., *Aircraft Design: A Conceptual Approach*, 3rd Ed., American Institute of Aeronautics and Astronautics, Reston VA., 1999.

Restar, S.A., Rogers, J.C., and Ronald, W.H., *Advanced Airframe Structural Materials, A Primer and Cost Estimating Methodology*, R-4016-AF.

$$Mfg.Labor = 73 \cdot 10.72 \cdot Weight^{0.82} \cdot Velocity^{0.484} \cdot Quantity^{0.641} \cdot M_L$$

$$Mfg.Material = 16 \cdot Weight^{0.921} \cdot Velocity^{0.621} \cdot Quantity^{0.799} \cdot M_M$$

$$ToolingLabor = 88 \cdot 8.71 \cdot Weight^{0.777} \cdot Velocity^{0.696} \cdot Quantity^{0.263} \cdot M_L$$

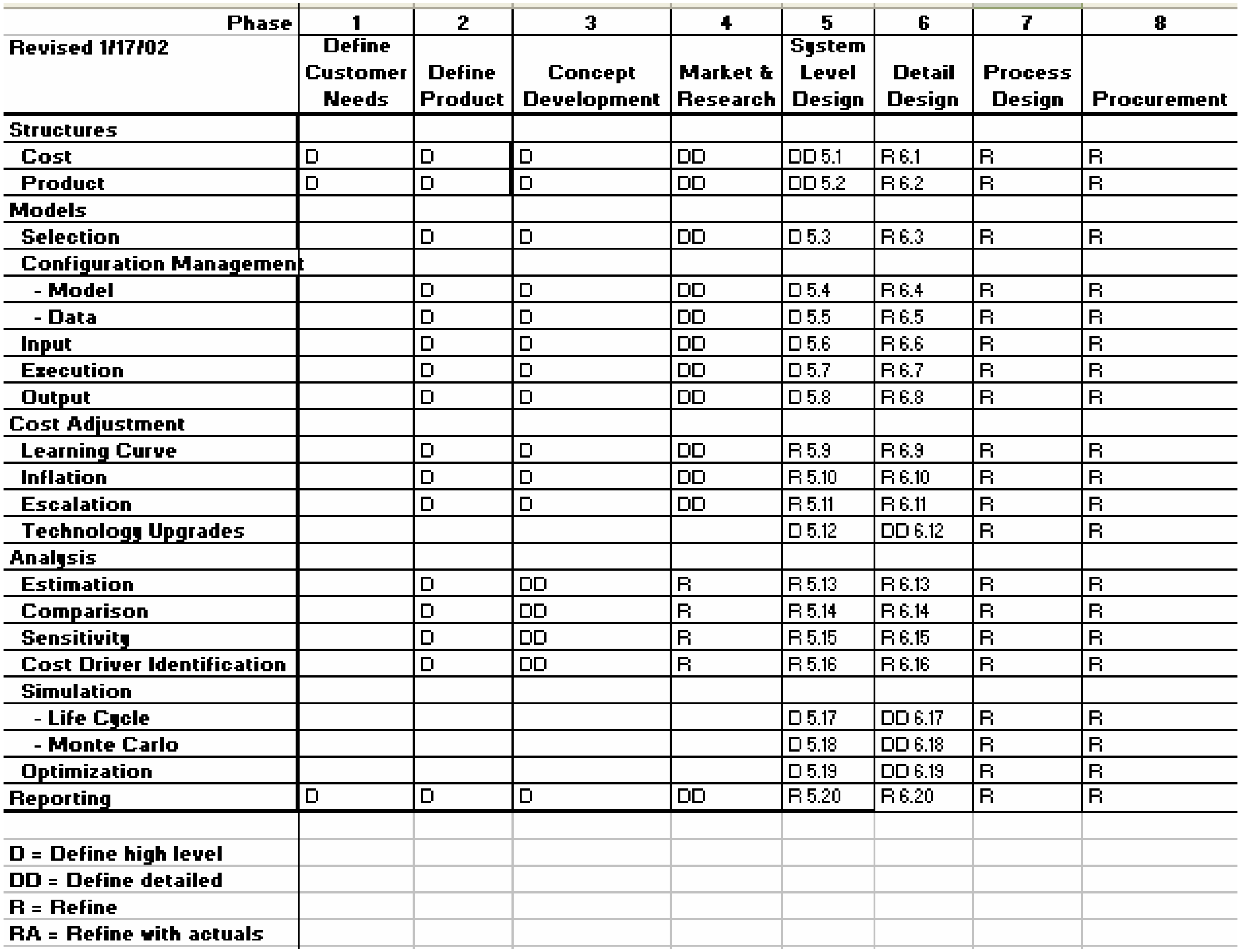
$$Eng.\&QCLabor = 86 \cdot 7.07 \cdot Weight^{0.777} \cdot Velocity^{0.894} \cdot Quantity^{0.163} \cdot M_L + 1.11 \cdot 0.133 \cdot Mfg.Labor$$

where;
M_L *= Material Labor Factor*
M_M *= Material Acquisition Factor.*
M_L is a man-hour complexity factor based on the type of material used. M_M is a material acquisition complexity factor based on the material type. This factor was obtained from [17]. The ranges for both M_L and M_M are provided in equations below.

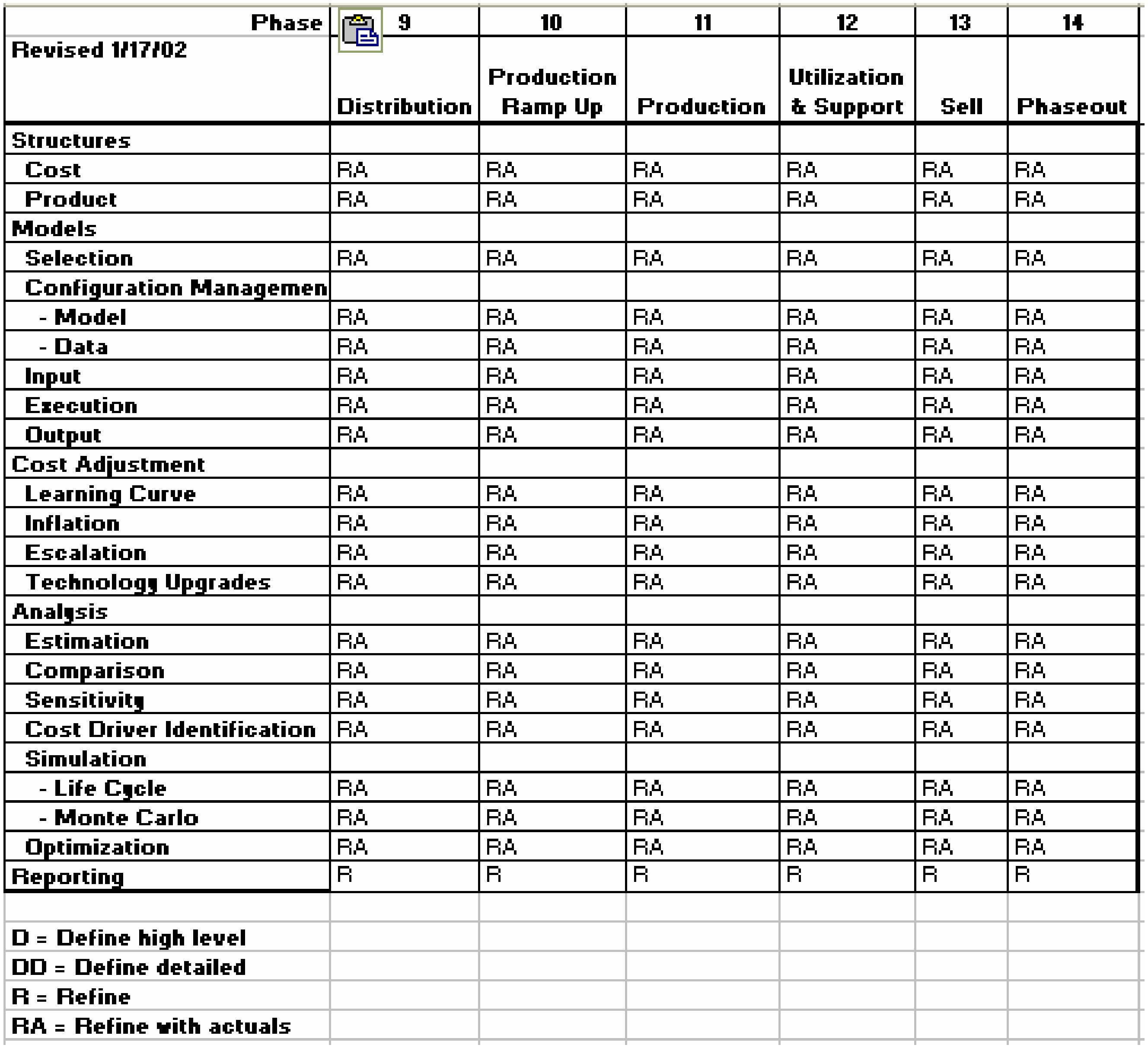
$$M_L = \begin{cases} 1.0\,for \quad Al \\ 1.1-1.8\,for \quad Composite \\ 1.5-2.0\,for \quad Steel \\ 1.3-2.0\,for \quad Ti \end{cases}$$

$$M_M = \begin{cases} 1.0\,for \quad Al \\ 5.05\,for \quad Composite \\ .82\,for \quad Steel \\ 3.27\,for \quad Ti \end{cases}$$

1.10.5 Appendix 1-E: Capabilities by Development Phase



Phase	1	2	3	4	5	6	7	8
Revised 1/17/02	Define Customer Needs	Define Product	Concept Development	Market & Research	System Level Design	Detail Design	Process Design	Procurement
Structures								
Cost	D	D	D	DD	DD 5.1	R 6.1	R	R
Product	D	D	D	DD	DD 5.2	R 6.2	R	R
Models								
Selection		D	D	DD	D 5.3	R 6.3	R	R
Configuration Management								
- Model		D	D	DD	D 5.4	R 6.4	R	R
- Data		D	D	DD	D 5.5	R 6.5	R	R
Input		D	D	DD	D 5.6	R 6.6	R	R
Execution		D	D	DD	D 5.7	R 6.7	R	R
Output		D	D	DD	D 5.8	R 6.8	R	R
Cost Adjustment								
Learning Curve		D	D	DD	R 5.9	R 6.9	R	R
Inflation		D	D	DD	R 5.10	R 6.10	R	R
Escalation		D	D	DD	R 5.11	R 6.11	R	R
Technology Upgrades					D 5.12	DD 6.12	R	R
Analysis								
Estimation		D	DD	R	R 5.13	R 6.13	R	R
Comparison		D	DD	R	R 5.14	R 6.14	R	R
Sensitivity		D	DD	R	R 5.15	R 6.15	R	R
Cost Driver Identification		D	DD	R	R 5.16	R 6.16	R	R
Simulation								
- Life Cycle					D 5.17	DD 6.17	R	R
- Monte Carlo					D 5.18	DD 6.18	R	R
Optimization					D 5.19	DD 6.19	R	R
Reporting	D	D	D	DD	R 5.20	R 6.20	R	R
D = Define high level								
DD = Define detailed								
R = Refine								
RA = Refine with actuals								



Phase	9	10	11	12	13	14
Revised 1/17/02	Distribution	Production Ramp Up	Production	Utilization & Support	Sell	Phaseout
Structures						
Cost	RA	RA	RA	RA	RA	RA
Product	RA	RA	RA	RA	RA	RA
Models						
Selection	RA	RA	RA	RA	RA	RA
Configuration Managemen						
- Model	RA	RA	RA	RA	RA	RA
- Data	RA	RA	RA	RA	RA	RA
Input	RA	RA	RA	RA	RA	RA
Execution	RA	RA	RA	RA	RA	RA
Output	RA	RA	RA	RA	RA	RA
Cost Adjustment						
Learning Curve	RA	RA	RA	RA	RA	RA
Inflation	RA	RA	RA	RA	RA	RA
Escalation	RA	RA	RA	RA	RA	RA
Technology Upgrades	RA	RA	RA	RA	RA	RA
Analysis						
Estimation	RA	RA	RA	RA	RA	RA
Comparison	RA	RA	RA	RA	RA	RA
Sensitivity	RA	RA	RA	RA	RA	RA
Cost Driver Identification	RA	RA	RA	RA	RA	RA
Simulation						
- Life Cycle	RA	RA	RA	RA	RA	RA
- Monte Carlo	RA	RA	RA	RA	RA	RA
Optimization	RA	RA	RA	RA	RA	RA
Reporting	R	R	R	R	R	R
D = Define high level						
DD = Define detailed						
R = Refine						
RA = Refine with actuals						

1.10.6 Appendix 1-F: Preliminary Capabilities List

1. Within the 'Life Cycle Costing Tool' the CBS elements of a particular program should be directly related to the WBS of the same program either by cross-indexing, coding of elements or actually being associated (attached) to a WBS component.[i (page 28)]
2. Within the 'Life Cycle Costing Tool', the Cost Breakdown Structure (CBS) elements of a particular program should be directly compatible through either cross indexing or coding of elements to the components of the following organizational reports and/or computer programs (legacy systems) for that particular program: (compatible = able to exist or act together harmoniously): [1 (page 28)]
    a. Scheduling networks
    b. Organization structure
    c. Planning documentation
    d. Work packages
3. Within the 'Life Cycle Costing Tool', the categories defined in the CBS should be coded in such a manner as to enable the separation of producer costs, supplier costs and consumer costs in an expeditious manner (to be able to produce aggregate costing reports). [1 (page 28)]
4. Within the 'Life Cycle Costing Tool', the CBS, and the categories defined, should be coded to facilitate the analysis of specific areas of interest independently (virtually ignoring) other areas. For example, you might want to investigate supply support costs as a function of engineering design, so you would want to be able to aggregate those costs together without other items in the report. [1 (page 28)]

Items 3 and 4 are similar, but accomplish two different things. They aggregate costs across different types of 'things'. Item 3 can aggregate the costs for a particular supplier over the whole product. Item 4 can aggregate the costs for a particular part (or sub assembly) in relation to a particular department of the company.

5. Hierarchies should be used to stratify the product and the cost. These are sometimes referred to as work breakdown structures and cost breakdown structures.
6. High Cost contributors:
    a. Need to be able to identify the high cost contributors for the different phases of the life cycle. For example, the high cost contributors of the initial production should be identified separately from the high cost contributors to maintenance and support. [1 (page 264)]
    b. Need to be able to identify the cause-effect relationships for the high cost contributors. That is, the system should identify the cause(s) of the high cost for each high cost contributor. [1 (page 28)]
    c. Need to be able to identify the specific input factor(s) that cause the item to be a high cost contributor. [1 (page 134)]
    d. Identify the criteria that the program uses to identify the high cost contributors in the different phases of the LCC.
7. Costs:
    a. Costs must be traceable to the specific input factor that causes the expense. [1 (page 264)]
    b. Utilizes an accepted cost evaluation approach.
        1. Possibly have options for using different approaches within the model for the different aspects of the model.
            1. Bottom up costing techniques
            2. Top-down costing techniques
            3. Backwards costing – using a budget cap as a constraint for the model.
        2. The program should guide the user on selecting the approach for a particular section of a model.

- c. Critical cost factors can be identified and using an accepted sensitivity analysis method, determine how sensitive the factors are to the cost analysis. [1] (page 270)
- d. The following cost factors should be included in the analysis structure for costs associated with:
    - 1. Design and development [1] (page 270)
    - 2. Production and testing [1] (page 270)
        - 1. Manufacturing labor [4] (page 407)
            - a. Separate the cost of engineering and design activities from those of manufacturing and assembly
            - b. Standard time data
                - i. Time study [4] (page 413)
                - ii. Predetermined time data (Maynard – MTM) [4] (page 419)
            - c. Direct Labor Cost
                - i. Rate schedules
                    - 1. Hourly
                    - 2. Salaried
                - ii. Set rates that represent job classifications relative to other jobs in the organization [4] (page 426)
            - d. Indirect labor Cost
                - i. Should include employer's contribution to employee [4] (page 427)
                    - 1. Social Security
                    - 2. Unemployment insurance
                    - 3. Worker's compensation insurance
                    - 4. Liability insurance
                    - 5. Health insurance
                    - 6. Pensions
                    - 7. Vacations
                    - 8. Holidays
                    - 9. Sick leave…etc.
                - ii. Calculated as a percentage applied to direct labor costs [4] (page 426)
        - 2. Material costs [4] (page 407)
            - a. Direct materials [4] (page 427)
                - i. Unit size plus finishing allowances (rough size of material needed to get the final product)
                - ii. Shrinkage
                - iii. Scrap from defective items
                - iv. Waste
            - b. Indirect materials [4] (page 427)
        - 3. Equipment and tooling
            - a. Allocation of capital cost over the life of the equipment and consider overhead directly applicable [4] (page 436)
            - b. Equipment cost include [4] (page 436-437)
                - i. Utilities
                - ii. Floor and space
                - iii. Maintenance and repair
            - c. Tooling cost include [4] (page 437)
                - i. Purchase cost (not amortized) or production cost for those that are made in-house
                - ii. Tool life considering the various machining conditions

            4. Supervision [4 (page 407)]
            5. Quality control and reliability and testing [4 (page 407)]
                a. Costs included [4 (page 440)]
                    i. Planning and implementation
                    ii. Inspection and testing
                    iii. Equipment
                    iv. Space
                    v. Utilities
                    vi. Special workplaces
                    vii. Supplies (forms)
                    viii. Personnel required to do the testing [4 (page 441)]
                        1. Wages
                        2. Overhead
                        3. Training for inspection personnel
                    ix. Cost of analyzing data
            6. Receiving and shipping [4 (page 407)]
            7. Packaging costs [4 (page 407)]
            8. Material handling and inventory costs [4 (page 407)]
            9. Distribution and marketing [4 (page 407)]
            10. Financing [4 (page 407)]
            11. Taxes and insurance [4 (page 407)]
            12. General and administrative [4 (page 407)]
            13. Plant overhead [4 (page 407)]
        3. Installation and checkout [1 (page 270)]
        4. Personnel training to include learning curve analysis [1 (page 270)]
        5. Technical data [1 (page 270)]
        6. Facility construction and maintenance [1 (page 270)]
        7. Spare and repair parts [1 (page 270)]
        8. Support equipment and tools [1 (page 270)]
        9. Inventory maintenance (post production) [1 (page 270)]
        10. Customer support (field service) [1 (page 270)]
        11. Program management [1 (page 270)]
    e. Should have the capability to use escalating techniques for determining future costs when needed (when the projected cost factors cannot be determined directly). [1 (page 270)]
    f. Should have the capability to use cost targets (CT): [1 (page 126)]
        1. Be able to allocate CTs to subsystems as design constraints
        2. Be able to evaluate alternative configurations in terms of the CTs
        3. LCC estimates should be compared to the initial CTs throughout the project
    g. Should include the following cost concepts: [1 (page 270)]
        1. Incremental and marginal costs
        2. Variable and fixed costs
        3. Discount rates
        4. Learning curves
    h. Cost aspects of all alternatives need to be treated in a consistent and comparable manner. [1 (page 270)]
    i. Cost amortization should be used appropriately. [1 (page 270)]
    j. Should have separate modeling of costs incurred during some defined initial period and over the remainder of the life cycle. [7]
8. General Execution Requirements for running the program:

a. The LCC tool should be have a modular configuration so that only the modules used in the model are executed when necessary. This would save time when executing the model. [ii]
b. Must be able to import data, whether linked to the data source or non-linked to the data source. [2]
c. Must be able to export data, whether linked to the data repository or non-linked to the data repository. [2]
d. Must be accessible through the web using acceptable methods for access and security. [2]
e. Must be easily modified when the model needs to be changed. [1 (page 264)]
f. Must execute in a timely manner. [1 (page 264)]
g. Must have acceptable error handling throughout program execution. [2]
h. The results of the model are repeatable. For example, two runs of the same configuration yield the same answer within a tolerable limit. [1 (page 264)]
i. Should have a methodology library. This library consists of modular blocks of previously developed sections of a model that can be reused in a new model. The user should be able to save their model sections in the library as needed. [2]
j. Should have an application program interface (API) for communication or rather transferring information between the tool and other systems. [2]
k. There should be a method of analyzing the tool's performance that uses acceptable techniques.
l. A version of the tool can be distributed to prime manufacturers or sub-system suppliers so they can provide highly focused and detailed information about their products in a format that permits timely comparative analysis.[iii]

9. General characteristics of LCC tool:
   a. The LCC tool should provide the environment to develop a comprehensive (valid) model that is reliable (repeatable). [1 (page 270)]
   b. Documentation of the specifics of the developed model can be produced automatically. [2]
      1. Ability to attach reference sources to the data
      2. Documentation can be traceable to a specific model configuration.

   c. Reporting
      1. Reports can be generated from the tool
      2. Data can be exported to other applications for manipulation and reporting
      3. Graphs and charts are possible [3 (page 23)]

   d. The model should have the capabilities to evaluate the following phases of the life cycle: [2]
      1. Research and development
      2. Production
      3. Operations and Support
      4. Retirement and disposal

   e. Risk Analysis
      1. Structured risk assessment for the interrelationships of various uncertainties [4 (page 268-269)]
         1. To identify the critical project elements
            a. A vehicle to determine the consequences of critical decisions
         2. To provide insight into those areas that increase the chance for a successful project
         3. Have the ability to bring out the inherent assumptions of the project
         4. To allow critical review to determine the validity of those assumptions
         5. Have the ability to quantify a large number of complex interrelated project uncertainties and define those uncertainties in terms of project risk

    6. Have the ability to track and forecast the success or failure of the project
2. Risk assessment of [iv (page 267)]
    1. Decision to undertake project
    2. Project alternatives
    3. Project cost
    4. Schedule
    5. Technical performance
3. Assessment of [4 (page 267)]
    1. Alternatives for risk reduction
    2. Project performance against the original risk assessment
4. Revised risk assessment based on project performance [4 (page 267)]
5. Cost risk [4 (page 267)]
    1. SAM
6. Project Schedule risk [4 (page 267)]
    1. PERT
    2. CPM
7. Risk Analysis should include
    1. Network analysis [4 (page 272)]
        a.
    2. Decision risk analysis
        a. Can use network analysis to represent the decision tree (with decisions and chance events) rather than a project structure [4 (page 272)]
            i. Takes into account the manager's pattern of performance [4 (page 300)]
            ii. Includes both internal and external interrelationships to the project [4 (page 300)]
        b. Decision tree development should include: [4 (page 300=301)]
            i. Identification of project objectives
            ii. Definition of the major decision points
            iii. Definition of the consequences oa each major decision
            iv. Identification of the probability of each consequence occurring
            v. Definition of the resolution of the consequences
            vi. Definition of the probability that the resolution will be successful
            vii. Definition of the cost or schedule impact of each problem resolution
        c. Must do more than just simulate the decision tree [4 (page 314)]
            i. Should define the management decision preference (goals and objectives, system performance requirements, generalized project approach)
            ii. Network construction should represent activities and events that lead to the defined preferences
    3. Cost risk analysis
        a. Not from the project structure but from the cost estimating structure [4 (page 272)]
        b. Risk associated with the individual cost elements [4 (page 272)]
        c. Areas of cost risk
            i. The independent variable uncertainty for the cost estimating relationship (CER) [4 (page 305)]

ii. Complexity factor uncertainty [4 (page 305)]
iii. CER statistical uncertainty [4 (page 305)]
iv. Design changes beyond those reflected by the ranges used for the CER independent variable [4 (page 311)]
v. Additional requirements beyond those represented by the cost line items [4 (page 311)]
vi. Risk associated with inappropriate design specifications, CERs, or complexity factors [4 (page 311)]
vii. Cost uncertainties internal to the individual WBS element such as the technical complexity of the item [4 (page 311)]
viii. Cost uncertainties external to the individual WBS element such as design changes, funding problems and work delays [4 (page 311)]

f. Sensitivity Analysis
   1. 'What if' analyses related to all types of potential risk [4 (page 509)]
      1. Estimating
      2. Configuration
      3. Schedule
   2. Examination of any implications of almost any aspect of the estimate
   3. The 'what if' questions should be anticipated when the structure of the estimate is first designed [4 (page 509)]

g. Should not allow sub-optimization without evaluating the overall affect of the sub-optimization to the total LCC.

h. Should be able to evaluate specific entities and / or overall factors and the relationships between them.

i. Assumptions within the LCC tool need to be reasonable. The model developer should know the assumptions. If assumptions are treated as fact when they are really uncertain, the developer needs to be aware when this happens. The assumptions should not treat quantitative or qualitative uncertainties for fact. [1 (page 270)]

j. Mixing the components within the model to get better alternatives should be possible. An optimization method would be good.

k. The reliability prediction methods should follow accepted methods of analysis for the different types of reliability requirements of the project. [2]

l. Uses accepted methods for statistical sampling.

m. Uses accepted methods for statistical analysis. For example, using the expected and average values correctly in the analysis of the model.
   1. Use accepted testing techniques among variables:
      1. To determine if data has significant relationships and the strength of those relationships (Correlation, Cross-tab, T-test)
      2. To determine if X actually yields Y (the affect of x on y) (ANOVA, MANOVA, ANCOVA)
      3. To determine the if Y can be predicted by X (Regression, Multiple regression)
      4. To determine if X can discriminate between levels of Y (Discriminate)

n. System Effectiveness
   1. Used to evaluate the system or product that is modeled to meet the overall operational demand within a given time when operated under specified conditions. [1 (page 270)]
   2. Uses an acceptable method to determine the effectiveness of the system or product developed within the LCC tool. System effectiveness can be related to

the ability of the system or product modeled to fulfill a defined need and is a function of performance, capacity, availability, readiness, reliability, maintainability, supportability, dependability, etc.

3. Has the capability to use several different measures of effectiveness and gives the user the choice of which to use. Could even recommend a method for a particular section of the model. When there are two or more effectiveness measures, the measures should be properly weighted in terms of significance or level of importance of each applicable criterion factor used in the model. [1 (page 270)]
4. The effectiveness measures need to be sensitive to changes in assumptions. [1 (page 270)]
5. The expected and average values that are used to determine the effectiveness of the model should be used correctly to measure the effectiveness. [1 (page 270)]
6. The effectiveness measures should be appropriate to the mission function. For example, the operational and maintenance concepts need to be adequately defined within the model and measured specifically to identify which iteration results of the model better fulfills the mission. [1 (page 270)]
7. The effectiveness measures and the cost aspects of alternatives should be treated consistently and should be comparable when evaluating different alternatives of the project. [1 (page 270)]
8. The cost and effectiveness factors should be linked logically so that when the cost changes the effectiveness factors change when necessary. [1 (page 270)]
9. The LCC tool uses the time value of money in its calculation of the future effectiveness of the system or product. [1 (page 270)]

o. Has simulation capabilities within the tool or can export data to a simulation package for analysis.
    1. Simulation can represent the dynamics of the life cycle

p. Should have the following capabilities for analyzing the solution during the Conceptual Design phase:
    1. Establish Cost Targets (CTs) for: [1 (page 134)]
        a. Unit acquisition
            i. Research and development
            ii. Production
        b. Operations and support
    2. Use CTs to design the best product

q. Should have the following capabilities for analyzing the solution during the Preliminary Design phase:
    1. Capable to perform trade-off analysis [1 (page 129)]
        1. Must be within CTs
        2. Must be overall cost effective (within the bounds set in the model)

r. Be capable to optimize solution for: [1 (page 134)]
    1. Overall objectives
    2. Sub sections of the solution (but only if overall optimization is also done)

s. Should have the following capabilities for analyzing the solution during the Detailed Design phase: [1 (page 270)]
    1. Evaluate the design characteristics
    2. Predict cost generating variables
    3. Estimate the cost
    4. Project the LCC as a cost profile
    5. Compare to initial requirements and Cost Targets

t. Should have the following capabilities for analyzing the solution during the production, utilization and support phase: [1 (page 270)]
    1. Be able to input data after implementation
    2. Re-evaluate the model with the actual data
    3. Assess the model with the new data and compare to original model
    4. Re-identify the high cost contributors and compare to original model

u. Assess the cause-effect relationships with the new data and compare to original model

v. Learning curve functions
    1. Should be an approximation that forecasts the cost reduction from one unit of production to its succeeding unit [4 (page 170)]
        1. Units of production are assumed to be essentially identical [4 (page 170)]
        2. The operation is repetitive [4 (page 170)]
    2. Should include an analysis on the effects of any work stoppage (the discontinuity of production)
    3. Should be based on the following:
        1. Number of units or subassemblies involved [4 (page 169)]
        2. Experience of the worker [4 (page 189)]
        3. The production rate [4 (page 169)]
        4. Type of design [4 (page 170)]
            a. New
            b. Existing with changes
        5. Program phase [4 (page 170)]
    4. Based on accepted learning theory [4 (page 170)]

w. Should have a ‘common core database’ or some other method that can fulfill the requirements for the various MIL-STD formats. Provide the interchange between the method used and the data exchange formats associated with the various Logistics Support Analysis (LSA) standards (MIL-STD). [3 (page 13)]

x. Can calculate the appropriate number of spare removable items (RI) to position at operating units and maintenance venues in order to achieve a desired level of operational availability and a high probability that RIs, required as resources for maintenance events, will be on hand when and where needed. [3 (page 23)]

y. Uses an acceptable method to determine the expected number of backorders (EBO) across the population of line replaceable units (LRU). The supply availability is a function of the EBO. The operational availability is the product of the maintenance availability and the supply availability. [3 (page 23)]

2. CONFERENCE/JOURNAL ARTICLES & TECHNICAL NOTES

Journal articles published:

2.1 Ormon, S. W., Cassady, C. R., and Greenwood, A. G. "Reliability Prediction Models to Support Conceptual Design," *IEEE Transactions on Reliability*, vol. 51, no. 2, June 2002, pp. 151-157.

Conference papers completed/published:

2.2 Greenwood, A. G. and Ormon, S. W. "A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk," *Proceedings of the Decision Sciences Institute Conference*, November 2002, San Diego.

2.3 Greenwood, A. G. and Ormon, S. W. "Development of a Generic Cost Estimation Process," completed; to be submitted for review to be presented at the *ASEM National Conference,* October 2003.

2.4 Ormon, S. W., and Greenwood, A. G. "A Comparison of Traditional and Object-Oriented Systems Analysis Tools," *10th Annual Industrial Engineering Research Conference (IERC)*, May 2001, Dallas.

Journal articles in preparation:

2-5 Greenwood, A. G. and Liaw, J. A. "A Framework for Engineering and Managing Requirements," in preparation for submission to *IEEE Transactions on Engineering Management*.

Technical Notes:

2.6 Ormon, S. W. and Greenwood, A. G. "The Use of Common Random Numbers with Monte Carlo Simulation."

2.7 Venugopalan, T. and Greenwood, A. G. "Simulation Based Design: Definition and Review."

2.8 Wu, T. C. and Greenwood, A. G. "Model Management: Definition and Review."

Sections 2.1 through 2.5 are published separately

2.6 TECHNICAL NOTE: The Use of Common Random Numbers with Monte Carlo Simulation

Stephen W. Ormon[1], Allen G. Greenwood[1], Ph.D., P.E.

Acknowledgement:
Support for this research was provided by the Air Force Research Laboratory and Mississippi State University.

A sensitivity analysis is conducted to study the affects of using common random numbers (CRN) in Monte-Carlo simulation. The analysis is conducted using CRT[1] and based on a high-level design of an airframe (typical of what may be considered during conceptual or preliminary design). The models used in the example are from [2] and [3].

Input data for the baseline case used in the example is provided in Table 1. The Quantity column is indented in order to depict the hierarchical relationships that exist within the P/WBS; e.g., there are 6 ribs in each wing and 2 wings for each airframe resulting in a total of 12 ribs. Input data are provided for, and cost estimates are performed for, only the components (leaves in the P/WBS); the other costs are “roll ups” of the component costs. For example, the wings cost is the sum of the ribs, skin, and spars costs.

Table 1. Baseline example inputs

Subsystem	Quantity	Acceptable Deviation from Mean (%)	Target Cost (millions)	Weight (lbs)	Material
Airframe	1	6	$3,025.0	--	--
Wings	2	5	$675.0	--	--
Ribs	6	2.5	$105.0	TRNG(15,20,25)	Aluminum
Skin	2	3	$400.0	220	Aluminum
Spar	2	5	$170.0	TRNG(70,75,78)	Aluminum
Fuselage	1	3	$980.0	1575	Aluminum
Air Inlet	1	5	$340.0	TRNG(250,300,350)	Titanium
LG	1	2	$845.0	--	--
Main	2	2	$600.0	TRNG(500,600,700)	Steel
Nose	1	5	$245.0	TRNG(170,175,180)	Steel
Tail	1	6	$300.0	TRNG(250,300,350)	Composite

[1] Department of Industrial Engineering, Mississippi State University

The results of the simulation, based on the data in Table 1, 1,000 replications, and a quantity of 500 aircraft, is provided in Table 2. This is considered the baseline case.

Table 2. Simulation results for airframe example -- Baseline

Subsystem	Quantity	Acceptable Deviation from Target (%)	Mean Deviation from Target (%) - Baseline	Target Cost (millions)	Expected Cost (millions) Baseline	Percentage Within Acceptable Range (%) - Baseline
Airframe	1	6	6	$3,025.0	$3,194.1	65.7
Wings	2	5	1	$675.0	$676.6	100.0
Ribs	6	2.5	4	$105.0	$105.4	34.0
Skin	2	3	1	$400.0	$405.6	100.0
Spar	2	5	3	$170.0	$165.6	99.0
Fuselage	1	3	1	$980.0	$968.2	100.0
Air Inlet	1	5	5	$340.0	$345.3	54.1
LG	1	2	5	$845.0	$886.8	15.6
Main	2	2	8	$600.0	$644.9	8.2
Nose	1	5	2	$245.0	$241.8	95.9
Tail	1	6	6	$300.0	$317.2	52.9

Alternative 1 modifies the design of the tail, which reduces the uncertainty of its weight; the tail's weight is set at a certain 290 pounds.

The simulation is executed for 1000 and 100 replications with the concept of CRN applied to each source of variation within the cost models parameters. Next, the simulation is executed for 1000 and 100 replications without applying the concept of CRN. Welch's difference-in-two-means t-test for significance between the baseline and Alternative 1 design are applied to each case. The results of the analysis are shown in Table 3.

Table 3a. Welch's Comparison of Baseline and Alternative #1 (CRN, 1000 Replications)

Using CRN - 1000 Replications

Subsystem	Expected Cost (millions) Baseline - $\bar{X}_B$	Standard Deviation of Cost (Millions) - $\bar{X}_B$	Expected Cost (millions) Alternative 1 - $\bar{X}_1$	Standard Deviation of Cost (Millions) - $\bar{X}_1$	Halflength	$\bar{X}_B$-$\bar{X}_1$	Significant at α=0.05
Airframe	$3,199.9	$38.8	**$3,193.6**	$36.7	$10.47	$6.3	no
Wings	$676.5	$5.8	$676.6	$5.8	$1.61	-$0.1	no
Ribs	$105.2	$5.6	$105.4	$5.6	$1.54	-$0.2	no
Skin	$405.5	$0.0	$405.6	$0.0	----------------	--------------	----------------
Spar	$165.7	$1.8	$165.6	$1.8	$0.50	$0.1	no
Fuselage	$968.2	$0.0	$968.2	$0.0	----------------	--------------	----------------
Air Inlet	$343.8	$22.6	$345.3	$22.6	$6.25	-$1.5	no
LG	$886.5	$23.8	$886.8	$23.8	$6.60	-$0.3	no
Main	$644.3	$23.1	$644.9	$23.1	$6.40	-$0.6	no
Nose	$242.2	$5.4	$241.8	$5.4	$1.49	$0.4	no
Tail	$325.0	$19.2	**$316.7**	$16.0	$4.89	$8.3	yes

Table 3b. Welch's Comparison of Baseline and Alternative #1 (CRN, 100 Replications)

Using CRN - 100 Replications

Subsystem	Expected Cost (millions) Baseline - $\bar{X}_B$	Standard Deviation of Cost (Millions) - $\bar{X}_B$	Expected Cost (millions) Alternative 1 - $\bar{X}_1$	Standard Deviation of Cost (Millions) - $\bar{X}_1$	Halflength	$\bar{X}_B$-$\bar{X}_1$	Significant at α=0.05
Airframe	**$3,197.8**	**38.5**	**$3,188.4**	**36.2**	$10.36	$9.4	no
Wings	$676.2	5.6	$676.2	5.6	$1.55	$0.0	no
Ribs	$105.2	**5.3**	$105.2	**5.3**	$1.47	$0.0	no
Skin	$405.6	0.0	$405.6	0.0	----------------	--------------	----------------
Spar	$165.5	1.7	$165.5	1.7	$0.47	$0.0	no
Fuselage	$968.2	0.0	$968.2	0.0	----------------	--------------	----------------
Air Inlet	$344.4	23.8	$344.4	23.8	$6.60	$0.0	no
LG	$884.4	24.9	$884.4	24.9	$6.90	$0.0	no
Main	$642.2	24.0	$642.2	24.0	$6.65	$0.0	no
Nose	$242.2	4.9	$242.2	4.9	$1.36	$0.0	no
Tail	**$324.6**	17.6	**$315.2**	14.4	$4.46	$9.4	yes

Table 3c. Welch's Comparison of Baseline and Alternative #1 (No CRN, 1000 Replications)

Without Using CRN - 1000 Replications

Subsystem	Expected Cost (millions) Baseline - $\bar{X}_B$	Standard Deviation of Cost (Millions) - $\bar{X}_B$	Expected Cost (millions) Alternative 1 - $\bar{X}_1$	Standard Deviation of Cost (Millions) - $\bar{X}_1$	Halflength	$\bar{X}_B$-$\bar{X}_1$	Significant at α=0.05
Airframe	$3,199.9	$39.2	**$3,189.4**	**37.8**	$10.67	$10.5	no
Wings	$676.5	$5.6	$676.9	5.9	$1.59	-$0.4	no
Ribs	$105.2	$5.4	$105.5	**5.6**	$1.52	-$0.3	no
Skin	$405.5	$0.0	$405.6	0.0	----------------	--------------	----------------
Spar	$165.7	$1.8	$165.8	1.8	$0.50	-$0.1	no
Fuselage	$968.2	$0.0	$968.2	0.0	----------------	--------------	----------------
Air Inlet	$343.8	$22.3	$344.5	23.2	$6.31	-$0.7	no
LG	$886.5	$24.2	$884.5	23.7	$6.64	$2.0	no
Main	$644.3	$23.3	$642.6	23.1	$6.43	$1.7	no
Nose	$242.2	$5.6	$241.9	5.5	$1.54	$0.3	no
Tail	$325.0	$19.8	**$315.4**	15.9	$4.98	$9.6	yes

Table 3d. Welch's Comparison of Baseline and Alternative #1 (No CRN, 100 Replications)

Without Using CRN - 100 Replications

Subsystem	Expected Cost (millions) Baseline - $\bar{X}_B$	Standard Deviation of Cost (Millions) - $\bar{X}_B$	Expected Cost (millions) Alternative 1 - $\bar{X}_1$	Standard Deviation of Cost (Millions) - $\bar{X}_1$	Halflength	$\bar{X}_B$-$\bar{X}_1$	Significant at α=0.05
Airframe	$3,202.4	$38.1	**$3,188.9**	**37.0**	$10.42	$13.5	yes
Wings	$676.6	$5.4	$677.8	5.9	$1.57	-$1.2	no
Ribs	$105.4	$5.4	$106.2	**5.7**	$1.54	-$0.8	no
Skin	$405.6	$0.0	$405.6	0.0	----------------	--------------	----------------
Spar	$165.6	$1.9	$165.9	1.9	$0.53	-$0.3	no
Fuselage	$968.2	$0.0	$968.2	0.0	----------------	--------------	----------------
Air Inlet	$345.6	$22.4	$342.7	21.7	$6.11	$2.9	no
LG	$885.8	$22.4	$885.0	24.7	$6.54	$0.8	no
Main	$644.5	$22.3	$643.7	24.3	$6.46	$0.8	no
Nose	$241.3	$5.2	$241.3	4.6	$1.36	$0.0	no
Tail	$326.3	$20.9	**$315.3**	16.3	$5.19	$11.0	yes

The simulation results that utilized CRN indicate the only significant difference in cost is for the airframe's tail. Also, it is important to note that the components that are not affected by the changes in the alternative design do not exhibit any changes in their component cost. Without applying CRN, the components that are not affected by the alternative design did exhibit changes in component cost due to the variability caused by the random number generator. Even though these differences were shown not to be significant (α=0.05), more complicated designs that require fewer simulation replications could exhibit higher variability of the component cost that is not attributed to actual design modifications. Although the airframe cost difference was significant by not applying CRN with 100 replications, the airframe cost difference was not significant using 1000 replications. One potential problem of applying CRN is that the number of unique random number streams utilized by the random number generator would increase dramatically with numerous sources of variation. Further research is needed to investigate the impact of not using CRN in a Monte-Carlo simulation.

REFERENCES

[1] Greenwood, A. G., and Ormon, S. W." A Hierarchical, Model-Based Approach and Tool for Estimating Cost Risk," *Proceedings of the 2002 Decision Sciences Institute Conference*, November 2002, San Diego,

[2] Raymer, D.P., *Aircraft Design: A Conceptual Approach*, 3rd Ed., American Institute of Aeronautics and Astronautics, Reston VA., 1999.

[3] Restar, S.A., Rogers, J.C., and Ronald, W.H., *Advanced Airframe Structural Materials, A Primer and Cost Estimating Methodology*, R-4016-AF, RAND Corporation.

2.7 TECHNICAL NOTE: Simulation-Based Design

Thiyagarajan Venugopalan [2], Allen G. Greenwood[1], Ph.D., P.E.

Acknowledgement:
Support for this research was provided by the Air Force Research Laboratory and Mississippi State University.

1. INTRODUCTION

The design process during the 1970s was generally based on a 'design-build- test' model and most of these steps were performed manually. Recently there has been a shift from the traditional engineering practices involving incremental design to a new design paradigm, which is greatly influenced by the use of computer modeling and simulation. This emerging field is referred to as simulation based design, computation based design, computational prototyping, multi-disciplinary design and optimization [1]. The simulation based design concept refers to simulation of the entire life cycle of the product from concept development to detailed design, prototyping, testing manufacturing operations, maintenance and disposal [2].

Simulation based design (SBD) and simulation based acquisition (SBA) define the process in both design and acquisition that utilizes the benefits of modeling and simulation (M & S) concepts and tools. The modeling and simulation tools are used in almost all defense and other projects in order to reduce the project cost and risk. With the use of modeling and simulation tools the problems are highlighted well before costs are committed thus leading to rapid exchange of concepts and data [3]. The SBD system is collaborative, multidisciplinary environment for developing and using virtual/ real prototypes. SBD allows to develop analyze and operate with virtual prototypes, as they would be actual physical prototypes without the cost and complexity involved with the real hardware and materials.

This report provides an analysis of the history of SBD and SBA, organizations involved in SBD, and some of the simulation-based tools that are available in the market. This report is a precursor for further research into the field of SBD and is organized as follows. Section 2 gives a brief overview about virtual prototyping (VP) and SBA. Section 3 provides an overview of the various organizations involved in SBD. Section 4 illustrates the various components of SBD. Section 5 gives an overview of managing the uncertainties in SBD, while Section 6 gives an insight into one of the simulation-based design optimization and finally Section 7 provides the conclusion and future research in SBD.

2. BASIC CONCEPTS IN MODELING AND SIMULATION

This section introduces some basic concepts such as virtual prototyping and simulation-based acquisition.

2.1 Virtual Prototyping

Virtual Prototyping (VP), also known as simulation-based design, refers to the iterative design refinement of a designed product using a computer-based functional physical simulation [15]. Many leading manufacturers such as Boeing, Chrysler and General Dynamics Electric Boat have saved millions of dollars by replacing the physical prototypes with the virtual prototypes or computer mock-ups [4].

[2] Department of Industrial Engineering, Mississippi State University

A virtual prototype has all of the properties and information that a physical prototype would have. The design, operability, production and testing of mechanical attributes of the product can be done before the product exists physically, thereby reducing the costs involved in production of the physical prototypes. Virtual prototype is a computer-based simulation of a system or subsystem with a degree of functional realism comparable to a physical prototype. The virtual prototyping environment addresses the engineering design concerns of the developer, process concerns of the manufacturer, logistical concerns of the maintainer and training and programmatic concerns of the operator. Virtual prototypes play a very active role in the new product development initiatives. With the aid of virtual prototypes and CAD/CAM tools products can be produced with a more robust design and shorter manufacturing cycle time. [5].

Virtual reality is the term that is often used interchangeably with virtual environment, telepresence, virtual prototyping, electronic environment, virtual factories, synthetic environments, scientific visualization, cyberspace, etc. Virtual reality consists of various levels of immersion. The most basic level is the non-immersive reality, which generally consists of a large projection system to provide a wide display for a large number of observers. It is used in the design environment for group design reviews. This level of virtual reality acts as an excellent training tool for many areas including manufacturing and process control. The intermediate level of virtual reality known as semi-immersive reality partially immerses the user within the environment. With this type of immersion the user remains aware of the surrounding real world environment. This is used as a tool to allow a large group to become a part of the virtual environment. The highest level of virtual reality is a fully immersive environment in which the individual is placed in a virtual environment that removes all reference to the surrounding real world. The individual becomes an active part of the virtual environment and has no visual reference to the surrounding real world. This type of immersion is obtained through the use of head mounted display (HMD), BOOM device, or a virtual reality CAVE [5].

The main benefits of virtual prototyping are greater acceleration in the production of new products, reduced cycle times which leads to new products being available in less time, and easier identification and management of risks [5].

2. 2 Simulation Based Acquisition (SBA)

SBA, now called Simulation and Modeling for Acquisition, Requirements, and Training (SMART), is a major initiative both within the Research Development and Acquisition (RDA) Modeling and Simulation (M&S) domain as well as the Department of Defense (DoD). It is intended to make appropriate use of M&S technologies in less time and at lower cost than traditional means [20].

Simulation Based Acquisition is a new systems acquisition paradigm that embraces the total system life cycle from initial realization of an unmet need, and carrying all the way forward through system retirement. This paradigm is supported by three principal components which are described below [21].

The first component is an evolved culture in which enterprise-wide and DoD-wide cooperation is the rule and in which individual technical contributions and innovations are encouraged and efficiently managed. This culture also encourages changes leading to enhanced concurrent development and the provision of incentives for organizations to provide tools and procedures for use by other programs. It recognizes and provides a means to encourage high-level performance versus affordability tradeoffs, both within a system as well as between systems, and without institutional or service imposed barriers.

The second component is a refined system acquisition process that capitalizes on changes in the acquisition culture engendered by SBA to facilitate collaboration by many Integrated Product Teams (IPT) across a system's entire acquisition life cycle.

The third component is an advanced SBA systems engineering environment in which the application of formal methods and automation to support all system life cycle activities simultaneously encourages software reuse and maximizes interoperability.

3. ORGANIZATIONS INVOLVED IN THE RESEARCH OF SIMULATION BASED DESIGN

This section provides a brief overview of the various organizations such as DARPA (Defense Advanced Research Project Agency), ARL(Applied Research Laboratory) and SBDC (Simulation Based Design Center) involved in the field of SBD. Many other organizations, such as Lockheed Martin Missiles and Space and Ball Aerospace and Technologies Corp, are involved in SBD; a complete list of SBD links is available in Ref. [14].

3.1 Simulation Based Design Program

DARPA plays a leading role in a number of Modeling and Simulation (M & S) programs. A very important and key project is the Simulation Based Design (SBD) program, which was initiated in 1993. The main goal of the project is to 'revolutionize the acquisition process for complex military and commercial products' using distributed, collaborative virtual development environments. SBD is considered to be multifaceted as it looks into all the aspects of the system acquisition process from mission analysis through design and logistics considerations to manufacturing and cost/risk analysis phases [6]. With the use of M & S the time and cost for all the areas of the acquisition process can be reduced. SBD seeks to integrate the technologies of distributed simulation, physics based modeling, and virtual environments [7].

The main objectives of the SBD program are [7]:

- reduce design time by half
- investigate advanced technologies "on-the-fly"
- eliminate physical prototypes
- improve initial design quality, resulting in significant life cycle cost reductions
- enhance communication using virtual reality technologies, giving a sense of experiencing the design
- assess manufacturing and operations, prior to construction, by using simulations

Applications of the SBD program include military and commercial ships, simulation based training, aviation/space systems, and electronics manufacturing and assembly. The development partners for the SBD program included Lockheed Martin, Newport News Shipbuilding, Science Applications International Corporation, General Dynamics Electric Boat Division, Deneb Robotics, Silicon Graphics Inc., and the Gulf Coast Region Maritime Technology Center.

The SBD program consists of two phases. Phase I addressed a ship hull mechanical/electrical problem and procedures for creating and refining CAD assembly models and related product data. The main challenge is to design and build a roll-on/roll-off ship that can transit a given distance faster than current ships. Phase II of the project, a $20M a year program begun in 1995develops and integrates critical technologies into a prototype system. The main goal is to create virtual prototypes of complex ships and military systems in a real time, interactive environment. Division Inc., a vendor of Interactive Product Simulation (IPS) tools, interfaced its dVISE, interactive software for designing and implementing virtual reality applications, with High Level Architecture (HLA), a Department of Defense (DOD) protocol for collaborative design [9]. SBD incorporates some basic HLA capabilities which allows virtual engineering prototypes to communicate with operational simulations and coordinate between multiple user activities with virtual prototypes that are composed of subsystems and parts.

The Graphical User Interface (GUI) for the SBD system was developed by Dramatis, a multimedia content development company. A familiar browser-like interface integrates common functions of location, querying, authoring and analyzing virtual prototypes. The Dramatis SBD workbench GUI is a Java application that runs on multiple platforms [10].

The SBD program performed a validation experiment in February 1997. The focus of this experiment was on survivability analysis and redesign of a surface combatant to meet a new threat. It included detailed physics-based models into the war fighting analysis phase and used multidisciplinary optimization techniques to provide parametric design information to the redesign process. The ASC experiment resulted in an SBD system configuration that [8]

- integrated multiple companies and government agencies into an Integrated Product Team (IPT)
- organized the IPT as a hierarchical collection of federations
- operated over a combination of local area network, internet and DARPA gigabit test bed network resources
- demonstrated the use of SBD in multiple life-cycle activities
- demonstrated the use of multidisciplinary analysis and optimization
- incorporated cost as an independent variable in the design trades

3.2 Simulation Based Design Center (SBDC)

The Gulf Coast Region Maritime Technology Center (GCRMTC) at the University of New Orleans operates the Simulation Based Design Center (SBDC) as a non-profit organization under the guidance of the Office of Naval Research. SBDC specifically focuses on integrating advanced information technologies and techniques to enhance design and engineering. The SBDC's virtual reality equipment and expertise make it an important resource for collaborative projects involving simulation-based design. The SBDC is involved in the field of enterprise modeling in which process simulation is performed to evaluate new technologies along with simulation and modeling of manufacturing processes, computer-based presentation of knowledge - virtual prototyping and immersive environment, product design, analysis methods and tools - which deal with life cycle cost analysis tools, methods for design and automation and knowledge capture and integration of design environments, simulation based design services – outreach, training, and visualization marketing support. SBDC is located in Houston, Texas and New Orleans, Louisiana. Research and development projects at the New Orleans site seek to validate and enhance virtual prototyping, while the Houston location assists companies that can benefit from the application of these technologies [5].

3.3 Applied Research Laboratory (ARL) and Simulation Based Design (SBD)

SBD acts as a key technology for product development and operation, as well as for efficient exploration of the conceptual design space for advanced systems. This process, and the resulting virtual prototypes, provides efficient technology transfer as well as early and accurate assessments of cost effectiveness and integrated life cycle support requirements. Applied Research Laboratory (ARL) at Pennsylvania State University has employed simulation-based design to achieve these objectives. The development of computer-aided design tools and object oriented, open software architectures, together with the conversion of model libraries to HLA compliant formats, has made the integration of design, performance prediction, and cost estimation toolsets possible. This integration in turn allows the construction of virtual prototypes of advanced systems and new system concepts using a simulation based design process. The resulting virtual prototypes are used to evaluate the cost effectiveness of new technologies in operationally realistic synthetic war games. The use of advanced tool kits for engineering analysis helps to achieve designs that are right the first time. Development costs are expected to be reduced by 30-50 percent, as documented in numerous industry studies [11].

The manufacturing technology department leads SBD process at ARL. The SBD system developed and implemented by ARL searches the design domain by integrating software tools for design creation, cost and performance estimation and object-oriented storage. This system utilizes virtual prototyping to rapidly evaluate design/decision alternatives for cost, mission effectiveness, reliability, maintainability, and manufacturability.

The SBD system shown in Figure 1 is an automated design process which captures rules for system and subsystem technologies, connects legacy applications over a distributed, heterogeneous environment, and helps engineers make decisions early in the acquisition process [12].

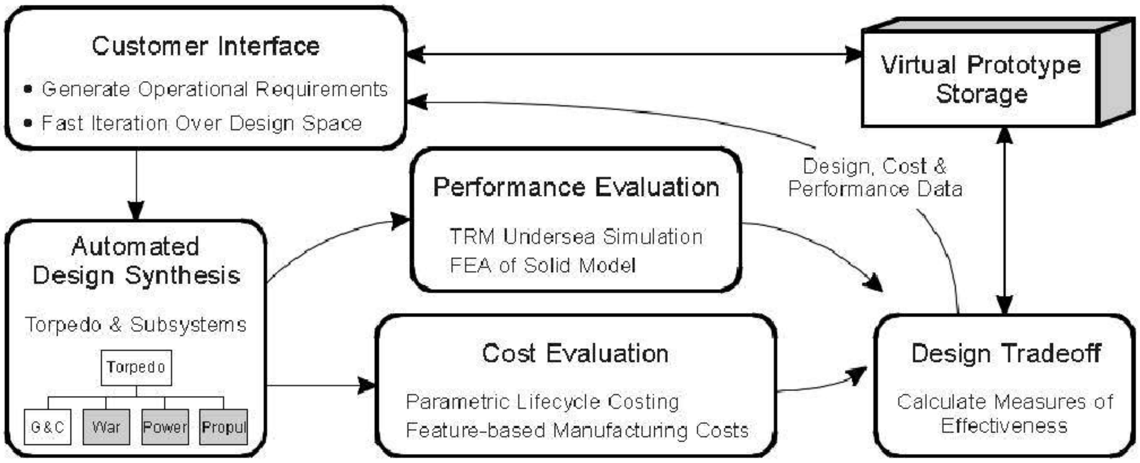

Customer Interface
• Generate Operational Requirements
• Fast Iteration Over Design Space
Virtual Prototype Storage
Design, Cost & Performance Data
Performance Evaluation
TRM Undersea Simulation
FEA of Solid Model
Automated Design Synthesis
Torpedo & Subsystems
Torpedo
G&C
War
Power
Propul
Cost Evaluation
Parametric Lifecycle Costing
Feature-based Manufacturing Costs
Design Tradeoff
Calculate Measures of Effectiveness

Figure 1: Simulation Based Design Flowchart

This SBD system interconnects knowledge based engineering tools, undersea simulations, computer aided design (CAD), commercial off-the-shelf (COTS) cost estimating tools, geography interpretational environment based on the Common Object Request Broker Architecture (CORBA), a middleware. SBD operates on various platforms including Windows, UNIX, and DEC Alpha VMS. The resulting virtual prototype uses geometry to show the relative sizing and placement of components and drive undersea simulations and parametric cost estimations. The infrastructure formalizes a natural dialog among agents. The information model defines the core for the SBD system by defining product and design variables such as vehicle length, diameter, speed, depth, propulsion power, weight and endurance. From this model,

a database is generated along with the CORBA interfaces to cost agents, performance agents, and design servers. Custom applications are developed as required with design servers capturing the rules that relate function and constraints to form. The design server checks input against the information model and then against the constraint map. A constraint map describes the possible combinations of inputs and outputs and ties the information model to the design server. As a result, a complete model is generated. Design model data are also used to determine product financial information [12].
The cost estimating system performs rapid financial evaluations of virtual prototypes through the commercial off-the-shelf (COTS) cost estimating software, PRICE Enterprise Edition. Data directly obtained from the design server generates a product cost estimate during the early design phase of the acquisition process. Hardware, software development, procurement and support costs are estimated parametrically. Sufficient cost, schedule and reliability information is provided for early conceptual design tradeoffs and proposals. By changing the design parameters, the SBD Cost Estimating System rapidly evaluates cost, schedule, and reliability for various product configurations. With this cost estimating system ARL can effectively evaluate 100 product configurations in an hour [13].

4. COMPONENTS OF SIMULATION BASED DESIGN

SBD is composed of the following components: modeling methods and computational tools, virtually reality environment, infrastructure for collaborative engineering, and integration technologies and tools [2].

4.1 Modeling methods and computation tools

The modeling methods include various modeling facilities and physics-based simulations that describes the behavior of the product during its entire lifecycle time. Many of these tools are available and provided through CAD/CAM/CAE systems. The innovative use of the computational physics distinguishes SBD tools from traditional design systems. Some of the SBD modeling tools available in the market are briefly discussed below.

CAE – Real time Object-oriented Simulation Environment (ROSE)
CAE Inc. offers a suite of products for modeling and simulation that focuses on real time, human-in-the-loop simulation for training and simulation based acquisition and design. SBA and SBD can optimize the functional performance and operational requirements of a new system. *ROSE* is a graphical modeling tool and real time simulation environment for developing simulations of complex systems. It is a complete SBD environment for the developing, testing, documenting, and controlling the simulation process. The main features of *ROSE* include real-time simulation based on first principles, automated compilation, linking and execution control, automated database generation, and an open architecture. [16].

Delmia Corporation
Delmia Corporation, previously known as Deneb Robotics Inc., provides 3D graphics-based factory simulation, telerobotic and virtual reality software. It provides a number of solutions in aerospace (airframe fabrication and assembly), shipbuilding (building strategy, fabrication, erection, outfitting, and modeling and simulation for mission scenarios and maintenance tasks) and the automotive industry to name a few. It is based on concurrent engineering with product engineering and process engineering occurring simultaneously through the Product, Process and Resource hub (DSPPR hub) [17].

PTC Inc.
PTC Inc. provides services for creating, sharing, distributing, and interacting with virtual products and environments. [18].

Electric Boat Corporation
Electric Boat Corporation – General Dynamics provides a simulation-based approach to surface ship and submarine designs [14].

Engineous Software Inc. - iSIGHT
iSIGHT automates the manual design processes associated with the development of new products and the redesign of existing products. It integrates simulation codes and provides engineering intelligence to drive the investigation of design alternatives thereby greatly reducing design cycle time and improvings product quality and reliability [19].

FakeSpace Inc.
FakeSpace Inc. provides design and manufacture interface devices that allow easy access to and manipulation of computer generated, three-dimensional visual simulations [14].

4.2 Virtual Reality Environment

The main objective of this component is to increase the interaction between the designers and developing products by blending virtual reality into the design environment. Designs can be created, modified, and visualized in real-time [2].

4.3 Infrastructure for Collaborative Engineering

This component provides a multimedia environment that enables collaborative design and integrated product viewing among teams at different locations.

4.4 Integration

A truly integrated design environment enables simulation and evaluation of several design alternatives and allows optimization of the product relative to the requirements and costs. The objectives of integration are consistency of data handling and simplicity of operation. IN developing such a system the following challenges have to be overcome namely the different user interfaces, no uniform data management approach in CAD, inconsistent meta-data capture, complicated assembly structures, data ownership disagreements, incompatibilities of multiple CAD systems. SBD is expected to evolve into a shared, highly flexible information based responsive design environment [2]

5. MANAGING THE EFFECT OF UNCERTAINTY IN SBD

It is generally recognized that there always exists uncertainties in any engineering systems due to variations in design conditions and mathematical models. Two general sources contribute to uncertainties in simulation prediction [13].

- External Uncertainty that comes from the variability in model prediction arising from plausible alternatives for input values. It is also called input parameter uncertainty.

- Internal Uncertainty comes from two sources:
    - model parameter uncertainty due to limited information in estimating the characteristics of model parameters for a given fixed model structure
    - model structure uncertainty that results uncertainty in the validity of assumptions underlying the model.

A critical issue in SBD is the effect of uncertainties of one subsystem or discipline that may propagate to another system through linked variables. The final system output will have the accumulated effect of the individual uncertainties. It is important to study the effect of various uncertainties as part of requirements tracking and design coordination. The two primary issues to be dealt with are (1) ways to propagate the effects of uncertainties across the subsystems and (2) to mitigate the effect of uncertainties and make reliable decisions.

There are two techniques for uncertainty analysis, extreme conditions approach, or worst-case analysis, and statistical approach [13].

The extreme conditions approach is used to derive the range of system output in terms of the range of uncertainties by either sub optimization or first-order Taylor expansion. This approach obtains an interval or the extremes of the final output from a chain of simulation models. The term extreme is defined as the minimum or maximum value of the end performance corresponding to ranges of internal and external uncertainties.

The statistical approach estimates the cumulative distributive function (CDF), probability distribution function (PDF), or population parameters of the final outputs from a chain of simulation models.

In most of the existing applications, the use of the extreme conditions approach and the statistical approach have been restricted to propagating the effect of external uncertainty and not internal uncertainty or the combination of the two. A recent development in design techniques has generated methods that can reduce the impact of potential variation by manipulating controllable design variables. To assist designers in making design decisions under uncertainty, propagating the effect of uncertainties is used to develop robust designs that mitigate the effects of both external and internal uncertainties. Robust designs are intended to make the system (or product) less sensitive to potential variations without eliminating the sources of uncertainties. [13].

6. SIMULATION-BASED DESIGN (SBD) OPTIMIZATION

Designers are confronted with the problem of finding settings for a large number of design parameters, called response parameters, that are optimal with respect to several simulated product or process characteristics. These parameters often originate from different engineering disciplines. The crucial question is how to find the best possible setting with a minimum number of simulations. Under these circumstances designers often use their intuition and experience. This can be improved by using statistical methods and optimization techniques. Stehouwer and Hertog describe a systematic non-iterative approach for design optimization using a tool called as *COMPACT* developed by CQM. With this method, as the design is improved, there is a better understanding of design parameters, a reduced number of simulations, and more time can be devoted to what-if-analyses [22].

The approach consists of following four steps [22]:

- Problem specification, which formulates the design optimization problem to find settings for the design parameters such that the design and response parameters satisfy certain constraints and the optimality requirement is satisfied.
- Design of experiments, which generates a set of suitably chosen design parameter settings or design points that must lie within the feasible design region.
- Compact modeling, which attempts to obtain good and compact model description for each of the response parameters in terms of the design parameters based on the results from step 2.
- Design Optimization. Steps 1 to 3 result in a Response Surface Model (RSM) for each of the response parameters that is used for prediction, optimization, and detailed analysis.

REFERENCES

[1] Swaminathan, S., Smidts, C, S., 1999, "A Framework for Assessing Confidence in Simulation Based Design under Uncertainty", *1999 Proceedings of Annual Reliability and Maintainability Symposium*

[2] Bossak, M, A., 1998, "Simulation Based Design", *Journal of Materials Processing Technology,* Vol. 76, pp 8-11

[3] Edwards, M., Nilsson, U., 2000, "Virtual Reality and Simulation Based Design- Fact or Fiction in Submarine Platform Design", http://www.subcorp.com.au/udt_epd3.pdf

[4] Puttur, M., 1998, "Simulation Based Design puts the Virtual World to Work", http://www.manufacturing.net/dn/index.asp?layout=articleWebzine&articleid=CA86063

[5] "Simulation Based Design Center", http://www.gcrmtc.org/sbdc/, Accessed on 11/21/2000

[6] "Challenges and Opportunities", http://www.dote.osd.mil/rr/acqchall.pdf

[7] Simulation Based Design Program, *MERIC Technical Briefs,* Number 14 http://www.uno.edu/~engr/meric/techbriefs/brief14.html, Accessed on 3/14/2001

[8] Jones, G., Graves, H., Gersh, M., 1997, "Integrated Acquisition-Logistics Synthetic Environments for Total Value Assessment", *PM,* pp 118-123

[9] "ARPA's Virtual Manufacturing Project", 1997, *Intelligent Manufacturing,* http://www.lionhrtpub.com/IM/Imsubs/IM-5-97/ARPA.html, Vol.14, No.5, Accessed on 05/17/2001

[10] "Simulation Based Design Workbench Graphical User Interface", *http://www.dramatis.com/sbd.html*, Accessed on 06/08/2001

[11] "Simulation Based Design", *http://www.arl.psu.edu/areas/simdesign/simdesign.html*, Accessed on 11/21/2000

[12] "Best Practice: Simulation Based Design Cost Estimating", *http://www.bmpcoe.org/bestpractices/internal/arlps/arlps_5.html*, Accessed on 05/18/2001

[13] Du, X., Chen, W., "A Methodology for Managing the Effect of Uncertainty in Simulation Based Design", *AIAA, http://www.uic.edu/labs/idel/pdf/AIAA_11_99.pdf*

[14] "Simulation Based Design Links", *http://www.nswcdc.navy.mil/mast/sbd_links.htm*, Accessed on 02/06/2003

[15] "Virtual Prototyping", *http://www.eng.buffalo.edu/Courses/MAE412/tutorials/W-introduction.html*, Accessed on 02/06/2003

[16] "Rapid Model Development and Runtime Environment (ROSE)", *http://www.cae.com/en/marine/pdf/rose.pdf*, Accessed on 02/07/03

[17] Delmia Corporation, *http://www.delmia.com*, Accessed on 02/07/2003

[18] PTC *http://www.division.com/appserver/mkt/solutions/index.jsp?s=0&k=289&l=92&l=93&l=94&l=95*, Accessed on 02/07/2003

[19] Engineous Software, Inc. *http://www.engineous.com/products.htm*, Accessed on 02/07/03

[20] "Definition of Simulation Based Acquisition (SBA)/ SMART", *http://www.stricom.army.mil/PRODUCTS/SBA/definition.jsp* , Accessed on 02/07/2003

[21] "SBA Functional Description", Version 1.1, *http://www.msiac.dmso.mil/sba_documents/intro-fi2.doc*, Accessed on 02/07/2003

[22] Stehouwer, P., and Hertog, D., 1999, "Simulation-Based Design Optimization: Methodology and Applications", *Proceedings of the First ASMO UK/ ISSMO Conference on Engineering Design Optimization, Ilkley, UK,*

2.8 TECHNICAL NOTE: Model Management: Definition and Review

Tai-Chi Wu[3], Allen G. Greenwood[1], Ph.D., P.E.

Acknowledgement:
Support for this research was provided by the Air Force Research Laboratory and Mississippi State University.

1. Introduction

Decision-making is a complex process of choosing a best alternative of activities in order to achieve a certain goal or objectives in an efficient manner. Because of the complexity of modern enterprises and manufacturing systems, problem-solving and decision-making process rely more and more on decision support tools or decision support systems (DSSs). Since models are a major component of DSSs, a good means to manage models is essential for successful decision-making. Section 2 provides a brief review of DSS; Section 3 describes MMS and its functionalities.

2. Decision support systems

A DSS includes computer-based technologies that are used to support decision-making and problem solving. Ginzberg and Stohr (1982) define a DSS as "a computer-based information system used to support decision-making activities in situations where it is not possible or not desirable to have an automated system perform the entire decision process" [GinM82]. DSSs are developed to deal with the structured portion of a problem, while the judgment of the decision-maker is still needed to deal with the unstructured part [ShiJ02]. This human-machine interaction is able to improve the decision-making process of semi-structured or unstructured decisions [ChaA93]. Zmud (1983) defines the tasks that a DSS should accomplish; a DSS must:

- capture and reflect what decision-makers (DM) think,
- be easy to use and learn,
- support multiple decision processes and multiple decision styles,
- help DMs to structure situations,
- help DMs in the initial stages of resolutions,
- allow a DM to adapt the system as he/she becomes more familiar with the system, and
- be user-friendly.

A DSS framework is used to set up the components in a DSS. DSSs may have different structures, such as BHW framework and SC framework [ChaA93]. A DSS that is constructed under the SC framework has three major components: a database, a model base, and a software system. The database holds data that is organized according to a specified schema that facilitates the DSS manipulating the data. The model base stores models or procedures which can be combined in different ways to generate new models for execution. The software system in the SC framework has three components: Database management software (DBMS), model base management software (MBMS), and dialog generation/management software (DGMS). DGMS is the interface between human and the DSS. It defines the requests a user can make and responses the DSS returns to the user.

The BHW framework is also composed of three components: Language system (LS), knowledge system (KS), and problem processing system (PPS). The LS defines the requests that the PPS can accept and acts on the KS. The KS contains the knowledge that can be used to generate responses to users' request through the PPS. In other words, the PPS processes user's requests (languages defined in LS), and

[3] Department of Industrial Engineering, Mississippi State University

according to user's requests, retrieves relevant pieces of knowledge (contents of KS), processes the knowledge pieces, and finally returns the solution.

Despite of different frameworks, all DSSs are composed of three main components: user interface (or dialogue generation and management software), database management system (DBMS), and model management system (MMS) [LenM93], [JelM85], [RizA98], [ShiJ02]. The user interface is the bridge between a DSS and users. It helps users formulate correct queries, processes the queries, directs the queries to the MMS and the DBMS, manages the execution of models and storage of result, and filters and presents the results to the users. The information flow may vary within different DSSs;however, the user interface plays an important controlling role.

The DBMS is software that helps define data schema, facilitates access, query, modify, update, filter, transform, and maintain data. Within a DSS, a DBMS is responsible for accessing data into the DSS in the format needed for execution, storing the results after execution, and providing data to the user interface upon request. The model management system assists model building and manipulation. One or more solvers may be included in a MMS to execute models. The solvers can also be the independent modules within a DSS. A MMS is responsible for managing model execution, this includes acquiring data from DBMS, scheduling model running, controlling input/output for models, etc. A more detailed discussion of model management system is provided below.

With the improvement in information technology, more sophistical tools are incorporated into DSSs, and thus increase the efficiency of decision-making and the effectiveness of the decisions [PeaJ95]. The most important information technology applied to DSS is the World Wide Web. By creating a Web-based DSS, it not only enables use regardless of geographical locations, but also connects data and DSSs more tightly [ShiJ02].

3. Model Management Systems

There are two types of "Model Management." The first type focuses on managing the process of model building. This type of model management divides into two levels. The first level involves monitoring the progress of model building for a project. The second level refers to the management of models across all projects. The models that used in this type of model management are not decision supporting models, they are used to document and understand the business process or the progress of projects. Examples of these types of models are: entity-relationship diagram, decomposition diagram, and data flow diagram [HudD93]. An entity-relationship diagram is a graphical representation of the entities, and the relationships between entities. Decomposition diagram is a top-down representation of functions of an enterprise or a system. A data flow diagram is a graphical representation of data stores, data process, and data flows. This paper focuses on the second type of model management, building and manipulating decision-support models.

Within a DSS, a model(s) may be composed of procedures, algorithms, or programs whereby data can be used or analyzed [ChaA93]. Models are simplified versions of real world systems, and help decision-makers conduct "what-if" experiments. Also, by the process of model building, users get to know a system better and may discover problems that would be overlooked without the rigorous definition required to build even the simplest models. Models are powerful components of DSSs, however; they are usually costly and time-consuming to build and maintain and oftentimes to use. The purpose of a MMS is to increase the productivity of the decision-making process [ChaA93]. Efficiently managing the models themselves and the process of building models improves the process of decision-making.

3.1 Definition of Model Management System

Model management systems are traditionally viewed as part of a DSS. A MMS is generally defined as a computer-based means that facilitates the development, storage, manipulation, control and effective utilization of models in an organization, as well to facilitate analysis [LiaT88], [ChaA93], [MuhW93]. Like data, models need to be managed. With the success of database management systems (DBMS), the earlier attempt of model base management system (MBMS) was to apply the similar approaches of DBMS to MBMS. However, model management is far more complex than data management. A lot of principles used in data management are not applicable for model management. Like DBMS, MBMS separates the users from the physical part of model processing. They all provide functions such as creation, modification, deletion, and maintenance. Unlike DBMS, models usually require sensitivity analysis [BlaR83]. Model selection is by far more complex then data query. MBMS may include one or more solvers for model execution while DBMS doesn't. Model creation is totally different from data entry.

Modern DSSs involve various types of knowledge (contained in, for example, simulation models, spreadsheets, and text documents). There is no single knowledge management technique powerful enough to handle all types of knowledge [ChaA93]. As for model management systems, there are various types of models, such as simulation models, and cost models. It is a major challenge to handle these disparate types of model in a single MBMS. While DBMS is a well-established research area, MBMSs are still at a relatively early stage of development[BlaR89].

There are three major functions in a MMS, they are [MuhW93]:

- Model development – activities include problem extraction, selection of modeling tools, model building, model validation and verification
- Model storage – activities include physical storage, metadata (logical view of models) storage, and model representation
- Model manipulation – activities include model selection, model integration, model instantiation, and model execution

Artificial intelligent (AI) has been introduced into MBMS. Examples of AI applications are model construction, model integration, model output interpretation, model construction helper, model validation and verification, natural language query processors for model bases, and integrations of model bases with other components of a DSS [BlaR89].

Even though MBMSs play a major role in DSSs, recent thought is to remove MBMSs from DSSs and consider it as an independent unit [BlaR89]. In order to increase the productivity and effectiveness of decision makers, modeling experts, and system developers, the needs of a MMS include [GeoA87], [MaJ97], [ChaA93]:

- standard formats for storing and presenting models,
- standard interfaces between models and model solvers,
- applicability at all the phases of the life cycle of decision-making,
- friendly user interfaces,
- provision of different views for different users,
- facilitate model development, including automatic model builders, such as wizards and templates,
- facilitate model reuse.

3.2 Model representation

There are different kinds of models, such as business models and physical models, but this paper focuses on decision-support models. A decision-support model is a simplified abstraction of a real-world situation that is built for analysis, learning, training, or problem-solving purpose [AskR93]. According to their degree of abstraction, decision models can be classified into three groups [TurE95]:

- Iconic (scale) models – a physical replication of the real system but usually at a different scale. This kind of models mimics everything in the system, logically and physically.
- Analog models – may not look like the real system physically, but tries to capture all the behaviors of the real system. This kind of models only focuses on the behaviors of the real world situation. It tries to capture every single element of logic part.
- Mathematical (quantitative) models – only capture the part of interest in a real world situation. Only behaviors that may have impact to the problems are of interest.

Among the three, mathematical models are most often used in DSSs because the complexity of the real system makes it impossible or too costly to represent in the other two formats.

As mentioned earlier, the views of a model vary considerably. There is no universal accepted way to build a model. Even the same model is represented in different way for various users or implemented differently in software. This can lead to problems in redundancy of model development and inconsistencies of models. Also, in order to work with models that are generated by different software, a user must learn more than one modeling skill. To reduce these problems, general modeling approaches are needed, e.g. the following model structures have been proposed.

- Relational approach – Blanning [BlaR89] attempts to implement the principle of relational views of data onto models. He views a model as a virtual file, where all possible inputs and corresponding output are records. For example, consider a virtual file called "a factory" (a factory model) where inside this file (a table) there are records with field names such as "order quantity" (input) and "shipped quantity" (output). By saving the input and output set, the differences between different models formats can be ignored. Thus the practices that are used in DBMSs can be applied to MMS with minimum changes. The Relational approach does not provide a standard format for models; instead, it tries to hide the physical part of models from users. This approach hides the model details from the decision maker and enables model selection by using query languages similar to SQL on input/output sets. This approach also helps model integration regardless of the physical difference of models.

- Structured modeling (SM)– Geoffrion [GeoA87] identifies the basic components of a model, the relationships between the components, and then represents a model as an acyclic graph that is composed of hierarchically organized elements (entities, attributes, and functions). Elements (primitive entity, compound entity, attribute, function, and test) are the basic units of the models. The relationship between each element is through "calls". All elements except primitive entity have a calling sequence that references to other elements. SM can capture the properties of a system that are independent of time as well as capture the numeric relationships between variables of mathematical models. Structured modeling decomposes a model into manageable elements. The arrangement of elements decides the functionality of a model. This approach provides a standard format for representing models. Model management becomes easier when all of the models are translated into one standard.

- Object-Oriented approach - An evolving model representation approach is to represent models using object-oriented (OO) principles [DolD86]. An object is an entity that has its own private data. The access to the data is via the methods or functions that the object provides. There are three kinds of models in object-orient modeling: static data model, dynamic model, and functional model/process model [MaJ98]. The static data model defines the objects and their attributes, as well as the static relationship between objects. The dynamic model defines the behaviors of objects over time, e.g. states, events, activities, and actions. The functional/process model defines how actions change the values of attributes or variables. The object-oriented approach has two main advantages. First, since models and model solvers can be modeled as objects, it is possible to represent various models within a single modeling paradigm. Second, the inheritability is promising for model reuse [KwoO96]. Structured modeling fails to describe the dynamic part of a simulation model in that SM can't describe the behaviors of entities in a certain point of time. While through OO approaches, the behavior of an entity (object) can be defined as an action of the entity (object).

It seems that the SM and OO approaches are more suitable to building a standard modeling language. In fact, there is a key similarity between SM and OO. "Structured modeling formalizes the notion of a definitional system as a way of describing models. This is precisely what the object-oriented concept of a class and the class-composition graphs formalize." [MuhW93] However, as mentioned in [GeoA99], SM is not suitable for representing simulation models without modification. SM is limited by its ability to capture the dynamic aspect of a simulation model.

3.3 Model Management Functions

MMS should provide functions to facilitate model manipulation and management. These functions include problem formulation, model creation, module libraries or templates, model execution, model storing, model deletion, documentation, model modification and updating, model selection, and model integration. These functions are used throughout the life cycle of decision-making. The steps in the life cycle of the decision making process are: [ChaA93]:

- problem stated,
- data collection or production,
- select reusable models or modules,
- integrate and modify models, or create a new model,
- link model with database if data input is needed during execution, and
- present and store solutions.

For "problem stated" and "data collection or production" steps, MBMSs may provide tools such as IDEF diagrams to help users formulate the problem and identify what data needs to be collected. Also, if data already exists in a DBMS, a MMS should help the users query the data from the DBMS. The MMS also needs to provide a documentation function to record the process of model building.

For "select reusable models or modules" step, MMSs should provide model libraries or templates. As for selecting existing models/modules, the MMS should provide selection functions to provide users with possible models/modules. Two approaches can be used in selection. The first is to select the existing models/modules from metadata or the contents of models/modules. The second is to select from their input/output schema, if input/output schema fits, then two models/modules may be able to connect. The second approach may not return the correct models/modules, but it's easier to implement. The first approach is harder to implement since it needs to recognize the knowledge of models/modules. Note human judgments are needed in either approach.

For "integrate and modify models, or create a new model" step, the MMS should facilitate the creation of models by providing modeling wizards, model libraries, templates, etc. MMSs usually contains well-developed model development software. However, a MMS may contain more than one model development tool, thus creating a problem of how to manage the compatibility of the different types models. Model integration is difficult task since models, especially from different software vendors, are usually not compatible. It is a difficult task to map the output schema of one model to another's input schema. Again, human judgments are needed in this step.

For “link model with database” step, MMSs should help acquire data dynamically at run time. MMSs should have the capability to communicate with DBMSs, select the right data and provide it to the models in the right format at the right time. Two approaches can be used. The first one is to select and organize data prior to a model’s execution. The second approach acquires data during model run time. The first approach is easier to implement and only slightly affects model execution. However, in some situations, the second approach is needed, especially when data are not available prior to a model’s execution.

For the “present and store solutions” step, it is known that the presentation of results to users is significant. MMSs should prepare the results after model execution for the user interface and DBMSs. MMS should also facilitate model storing and updating as users conduct “what-if” experiments.

Without a standard model representation, high-level model manipulation functions, such as model integration and model selection, are difficult [HarI84]. The lack of a standard modeling format results in such difficulties as a lack of modular construction, differing computer languages, and a lack of standardization of input and output structures. A standard modeling format also facilitates model integration. In DBMS, SQL is a recognized standard language to select data. In MMS, however, there is no standard language available to select a model since different model bases have different ways to define its models. Artificial intelligence (AI) has been used in model selection to recognize the reasoning knowledge of model definition rules. [KwoO96] Still, model selection is a difficult task and more research is needed.

Model reuse is an attempt to utilize the whole or part of existing models when building a new model. The main advantage of reusing of models is saving model building time and effort. Since models may be built using different approaches, interpretation of models and then re-structuring them into a manageable format is required. There are two general approaches to model reuse. The first is to design reusable model constructs, for example, a model library or model template. The second approach is to use a specific modeling language, or translate models into a standard modeling representation [KwoO96]. An object-oriented approach largely increases the reusability of models. However, a modeling standard is still needed to ensure model reusability [RizA98]. Kown and Park (1996) proposed a reverse modeling (RM) approach to reuse models. In their approach, they assume that each model has metadata that defines the model name, type of model, purpose of model, and modeling language. After identifying the model, constructs (elements) and their relationships are extracted. This process is done by a specific translator, that is, for each type of model, there should be a corresponding translator to transform the model constructs and relationships into a standard format. Models in the standard format then can be reused.

For models that are built in a closed environment, a translator is needed when they are to be imported/exported to another environment. For example, simulation models built with *ProModel* are not readable by *Witness*. For complex models, especially in a cooperative model building environment, extensive model sharing and exchange capabilities are needed. This requires at least one translator among various formats[KimH01]. In most cases, no such translator exists. Thus, an open-architecture model-exchange environment is needed. Currently, an evolving research area is the use of XML (eXtensible Markup Language) to build such an environment. XML was originally designed to support electronic publishing and has the following characteristics:

- simplicity – an XML document is easy to read and edit,
- extensibility – the format of XML documents can be easily extended to include more data,
- interoperability – XML is widely accept and works on various platform and software,
- openness – an XML document is easy to read and edit and plays an increasingly important role in various kinds of data exchange.

XML provides a natural way to design open-architecture models.

References

[BlaR82] Blanning, W. R, “Data Management And Model Management,” Proceeding of the ACM Twentieth Annual Southeast Regional Conference, April 1982, pp. 139-147.

[MuhW93]Waleed, A. W., “An object-oriented framework for model management and DSS development,” Decision Support Systems, Vol. 9, 1993, pp. 217-229

[ChaA93] Chang, A., Holsapple, C. W., Whinston, A. B., “Model management issues and directions,” Decision Support Systems, Vol. 9, 1993, pp. 19-37.

[ShiJ02] Shim, J.P., Warkentin, M., Courtney, J.F., Power, D.J., Sharda, R., and Carlsson, C., “Past, present, and future of decision support technology,” Decision Support Systems, Vol. 33, 2002, pp. 111-126.

[PeaJ95] Pearson, J.M., Shim, J.P., “An empirical investigation into DSS structures and environments,” Decision Support Systems, Vol. 13, 1995, pp. 141-158.

[GinM82] Ginzberg, M.J., and Stohr, E.A., “Decision support systems: issues and perspectives,” In Ginzberg, M.J., Reitman, W., and Stohr, E.A., eds. Decision Support Systems, Amsterdam: North-Holland, 1982, pp. 9-31.

[ChaA93] Chang, A., Holsapple, C. W., and Whinston, A. B., “Model Management Issues and Directions,” Decision Support Systems, Vol. 9, No. 1, 1993, pp. 19-37.

[ZmuR83] Zmud, R.W., Information Systems in Organizations, Glenview, IL: Scott, Foreman, 1983.

[LenM93] Lenard, M.L., “An object-oriented approach to model management,” Decision Support System, Vol. 9, 1993, pp. 67-73.

[JelM85] Jelassi, M.T, Jarke, M., and Stohr, E.A., “Designing a Generalized Multiple Criteria Decision Support System,” Journal of Management Information Systems, Vol. 1, No. 4, Spring 1985,pp. 24-43.

[RizA98] Rizzoli, A.E., Davis, J.R., and Abel, D.J., “Model and data integration and re-use in environmental decision support systems,” Decision Support Systems, Vol. 24, 1998, pp. 127-144.

[HudD93] Hudson, D.L., Practical Model Management Using CASE Tools, QED Publishing Group, Boston, 1993.

[LiaT88] Liang, T. P., “Development of a Knowledge-Based Model Management System,” Operations Research, Vol. 36, No. 6, November-December 1988, pp. 849-863.

[BlaR89] Blanning, R. W., “Model Management Systems: An Overview,” Owen Graduate School of Management, Vanderbilt University, Working Paper No. 89-23, November 1989.

[GeoA87] Geoffrion, A. M., “An Introduction to Structured Modeling,” Management Science, Vol. 33, No. 5, May 1987, pp. 547-588.

[MaJ97] Ma, J., “Type and inheritance theory for model management,” Decision Support Systems, Vol. 19, No. 1, 1997, pp. 53-60.

[AskR93] Askin, R.G., Standridge, C.R., Modeling and Analysis of Manufacturing Systems, John Wiley & Sons, Inc., New York, 1993.

[TurE95] Turban, E., Decision Support and Expert Systems: Management Support Systems, Prentice-Hall, inc., NJ, 1995.

[CheY88] Chen, Y. S., “An Entity-Relationship Approach to Decision Support and Export Systems,” Decision Support Systems, Vol. 4, 1988, pp. 225-234.

[DokD86] Dolk, D.R. “Data as Models: An Approach to Implementing Model Management,” Decision Support Systems, Vol. 2, 1986, pp. 73-80.

[KwoO96] Kwon, O Byung, Park, Sung Joo, “RMT, A modeling support system for model reuse,” Decision Support Systems, Vol. 16, 1996, pp. 131-153.

[GeoA99] Geoffrion, A.M., “Structured Modeling: Survey and Future Research Directions,” http://www.anderson.ucla.edu/faculty/art.geoffrion/home/csts/index.htm, June 1, 1999.

[HarI94] Harvey, I. T., Grant, P. W., and Jobling, C. P., “Model Management for Computer-Aided Control Engineering in an Object-Oriented Environment,” Conference publication, Vol. 389, Issue: 1, No. 389, 21-24 March, 1994, pp. 431-436.

[RizA98] Rizzoli, A.E., Davis, J.R., and Abel, D.J., “Model and data integration and re-use in environmental decision support systems,” Decision Support Systems, Vol. 24, 1998, pp. 127-144.

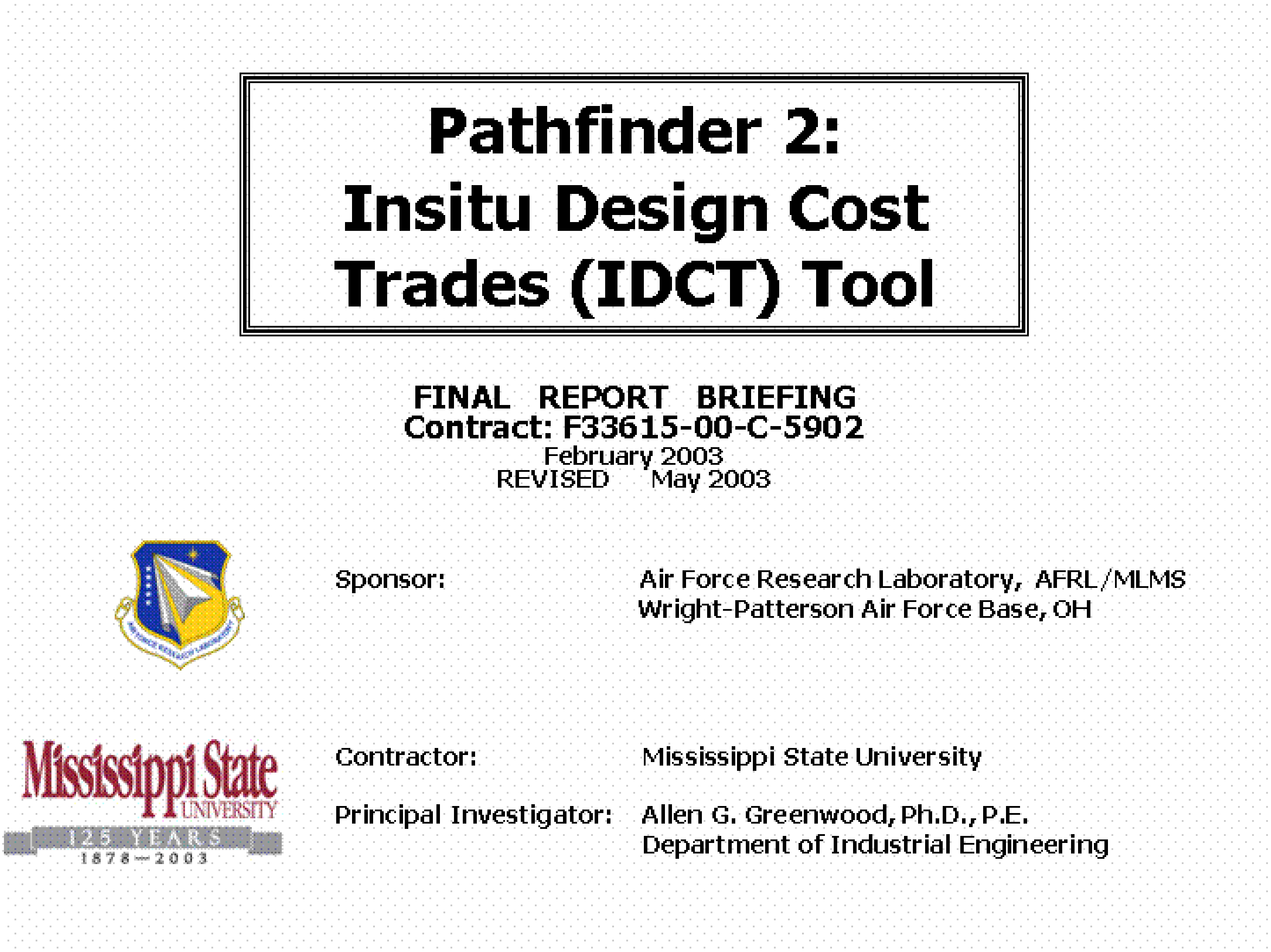
[KimH01] Kim, H., “An XML-based modeling language for the open interchange of decision models,” Decision Support Systems, Vol. 31, No. 4, October 2001, pp. 429-441.

[MaJ98] Ma, J., Tian, Q., Zhou, D., “An object-oriented approach to structured modeling,” RMA '98 Proceedings, 1998 Information Resources Management Association International Conference, May 1998, pp. 406-413.

3. BRIEFING CHARTS

The following table maps each briefing chart to descriptions in Sections I and II where information contained in the chart is discussed in more detail.

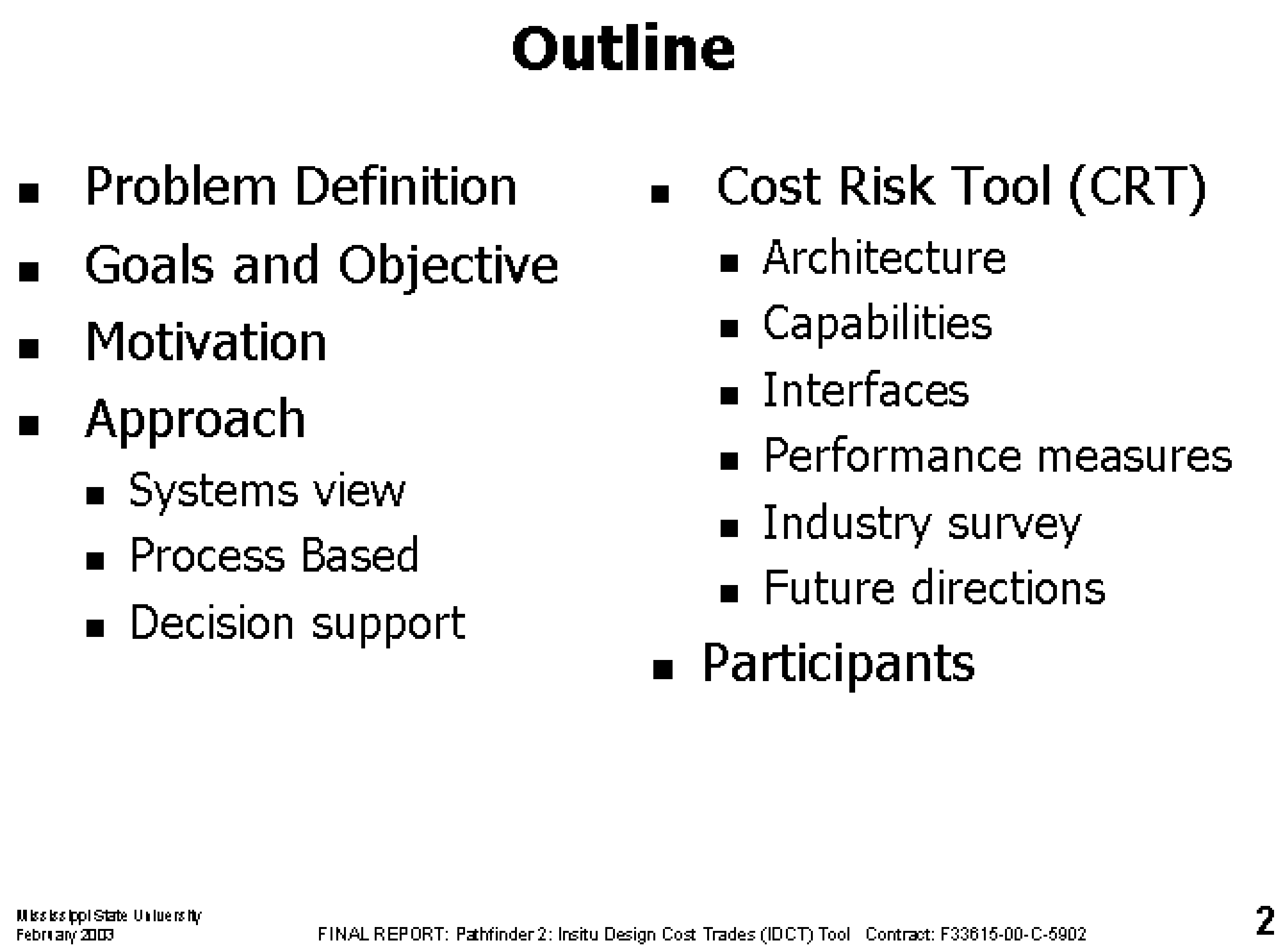
Briefing Chart Number	Section reference in text
1	N/A, title chart
2	N/A, outline of presentation
3	1.1
4	1.2
5	1.5
6	1.3
7	1.3
8	1.3
9	1.5.3
10	1.5.3
11	1.5.3
12	1.5.3
13	1.3
14	1.5.1
15	N/A, summary chart
16	1.6.2
17	1.6.3
18	1.6.3
19	1.6.4
20	1.6.4
21	1.6.4
22	1.6.4
23	1.6.4
24	1.6.4, 1.6.5
25	1.6.5
26	1.6.5
27	1.6.5
28	1.6.4
29	1.6.4
30	1.5.2.4
31	1.6.4
32	1.5.2.3, 1.6.1, 1.6.3
33	1.6.6
34	1.7



Pathfinder 2: Insitu Design Cost Trades (IDCT) Tool

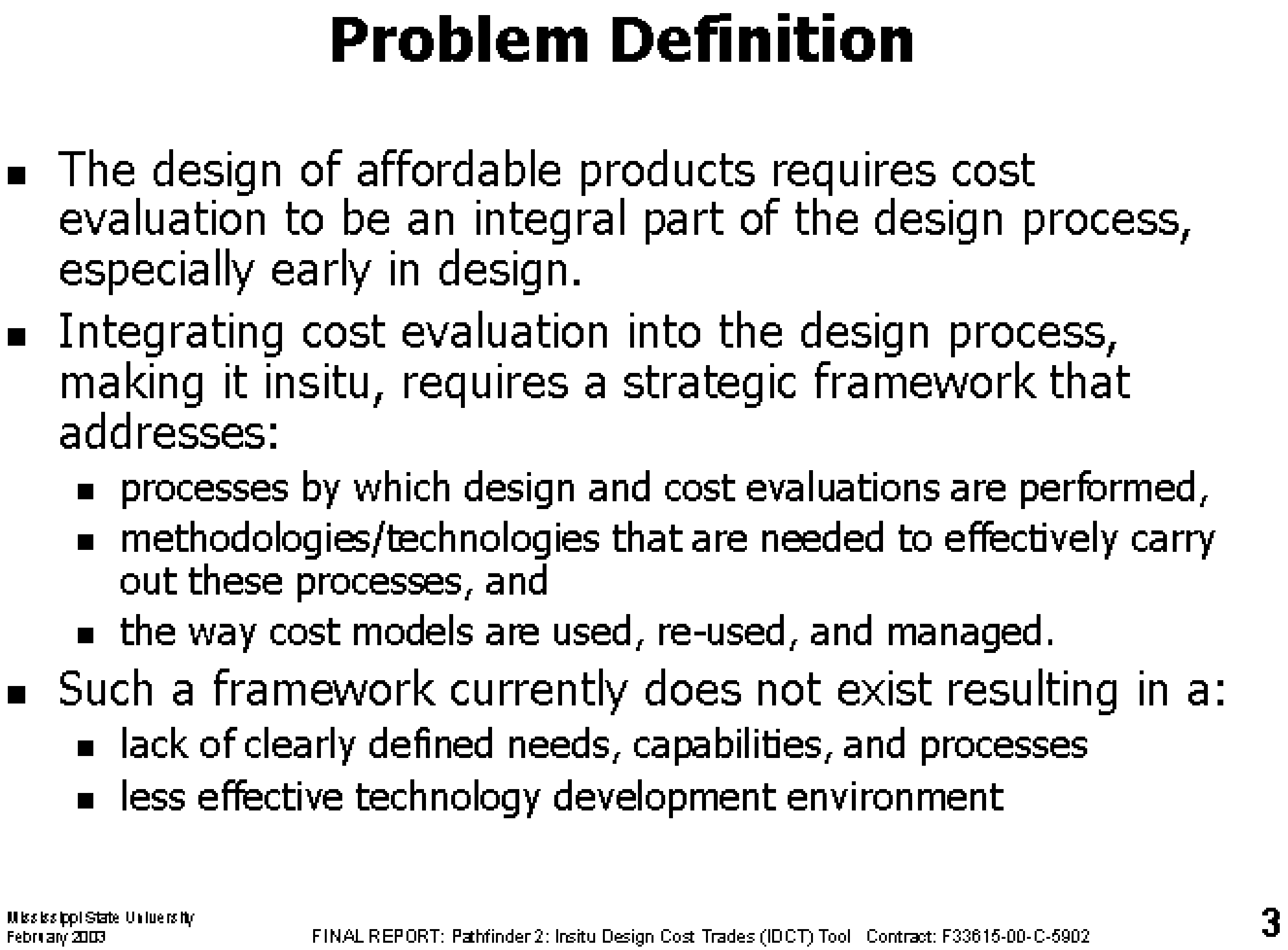
FINAL REPORT BRIEFING
Contract: F33615-00-C-5902
February 2003
REVISED May 2003

Sponsor:	Air Force Research Laboratory, AFRL/MLMS Wright-Patterson Air Force Base, OH
Contractor:	Mississippi State University
Principal Investigator:	Allen G. Greenwood, Ph.D., P.E. Department of Industrial Engineering



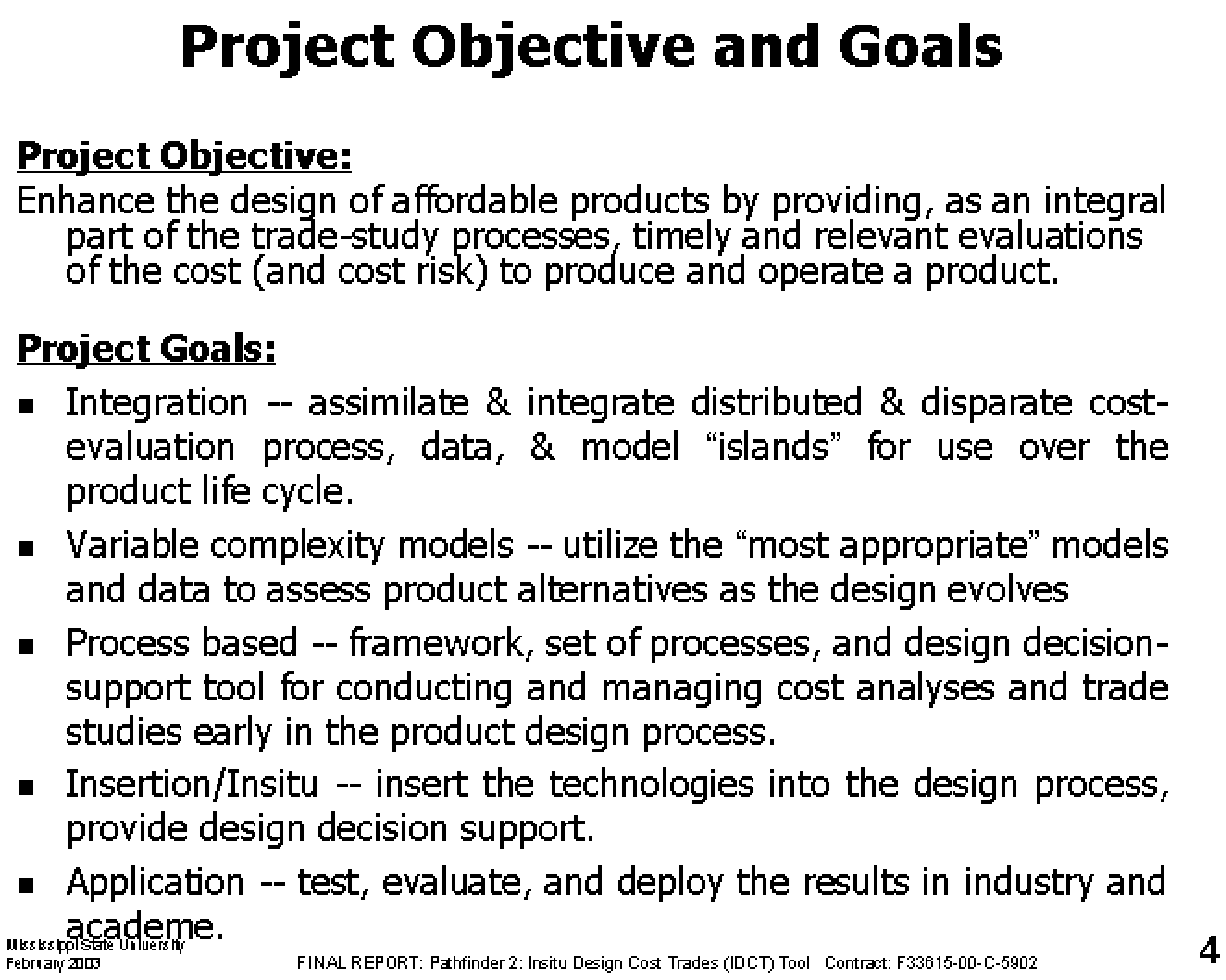
Outline

- Problem Definition
- Goals and Objective
- Motivation
- Approach
  - Systems view
  - Process Based
  - Decision support
- Cost Risk Tool (CRT)
  - Architecture
  - Capabilities
  - Interfaces
  - Performance measures
  - Industry survey
  - Future directions
- Participants



Problem Definition

- The design of affordable products requires cost evaluation to be an integral part of the design process, especially early in design.
- Integrating cost evaluation into the design process, making it insitu, requires a strategic framework that addresses:
  - processes by which design and cost evaluations are performed,
  - methodologies/technologies that are needed to effectively carry out these processes, and
  - the way cost models are used, re-used, and managed.
- Such a framework currently does not exist resulting in a:
  - lack of clearly defined needs, capabilities, and processes
  - less effective technology development environment



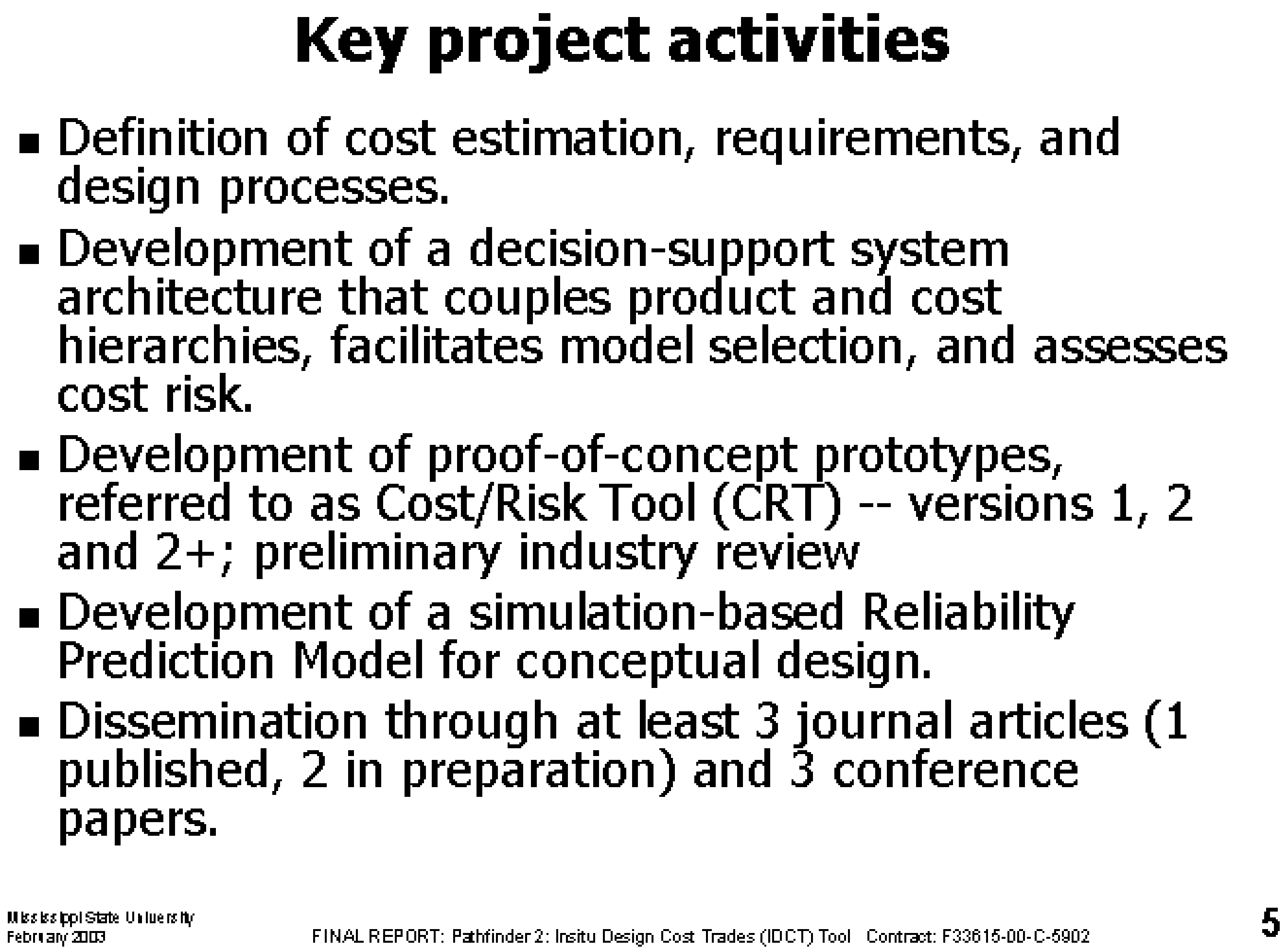
Project Objective and Goals

Project Objective:

Enhance the design of affordable products by providing, as an integral part of the trade-study processes, timely and relevant evaluations of the cost (and cost risk) to produce and operate a product.

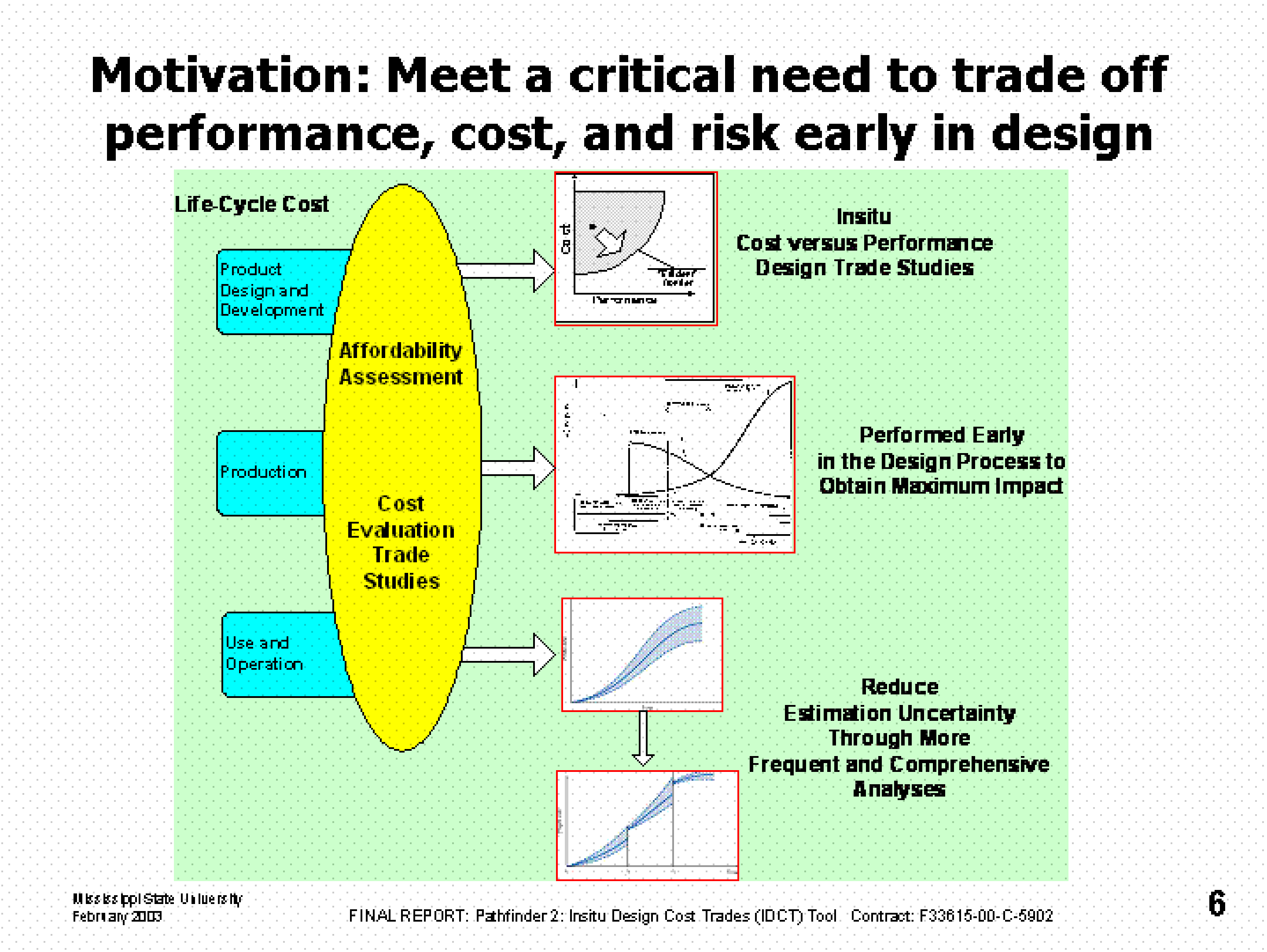
Project Goals:

- Integration -- assimilate & integrate distributed & disparate cost-evaluation process, data, & model "islands" for use over the product life cycle.
- Variable complexity models -- utilize the "most appropriate" models and data to assess product alternatives as the design evolves
- Process based -- framework, set of processes, and design decision-support tool for conducting and managing cost analyses and trade studies early in the product design process.
- Insertion/Insitu -- insert the technologies into the design process, provide design decision support.
- Application -- test, evaluate, and deploy the results in industry and academe.



Key project activities

- Definition of cost estimation, requirements, and design processes.
- Development of a decision-support system architecture that couples product and cost hierarchies, facilitates model selection, and assesses cost risk.
- Development of proof-of-concept prototypes, referred to as Cost/Risk Tool (CRT) -- versions 1, 2 and 2+; preliminary industry review
- Development of a simulation-based Reliability Prediction Model for conceptual design.
- Dissemination through at least 3 journal articles (1 published, 2 in preparation) and 3 conference papers.



Motivation: Meet a critical need to trade off performance, cost, and risk early in design

Insitu Cost versus Performance Design Trade Studies

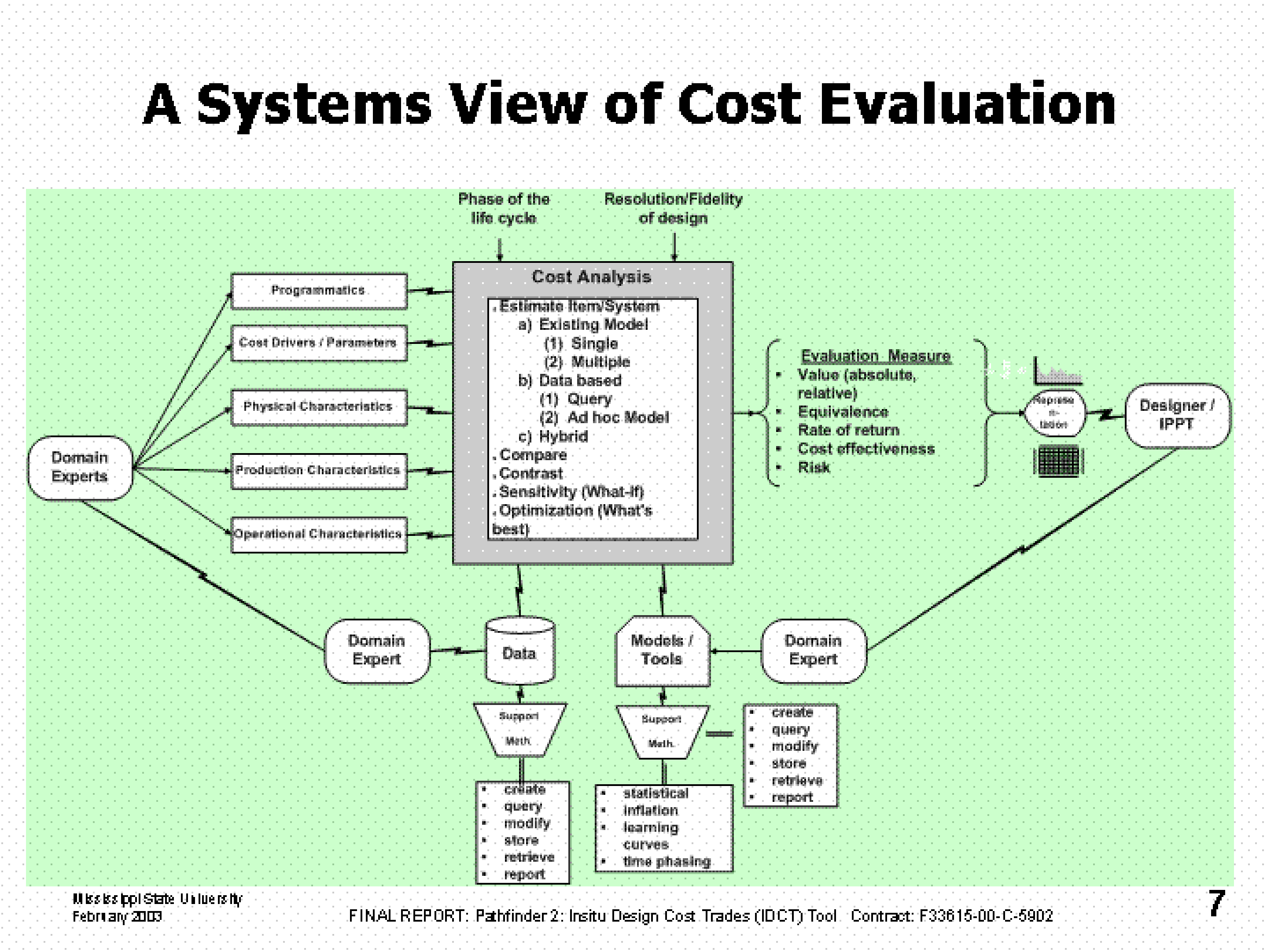
Performed Early in the Design Process to Obtain Maximum Impact

Reduce Estimation Uncertainty Through More Frequent and Comprehensive Analyses

Mississippi State University
February 2003

FINAL REPORT: Pathfinder 2: Insitu Design Cost Trades (IDCT) Tool Contract: F33615-00-C-5902

6



A Systems View of Cost Evaluation

Phase of the life cycle

Resolution/Fidelity of design

Cost Analysis

- Estimate Item/System
  - a) Existing Model
    - (1) Single
    - (2) Multiple
  - b) Data based
    - (1) Query
    - (2) Ad hoc Model
  - c) Hybrid
- Compare
- Contrast
- Sensitivity (What-if)
- Optimization (What's best)

Programmatics

Cost Drivers / Parameters

Physical Characteristics

Production Characteristics

Operational Characteristics

Domain Experts

Evaluation Measure

- Value (absolute, relative)
- Equivalence
- Rate of return
- Cost effectiveness
- Risk

Represe
n-
tation

Designer / IPPT

Domain Expert

Data

Support Meth.

- create
- query
- modify
- store
- retrieve
- report

Models / Tools

Domain Expert

Support Meth.

- statistical
- inflation
- learning curves
- time phasing

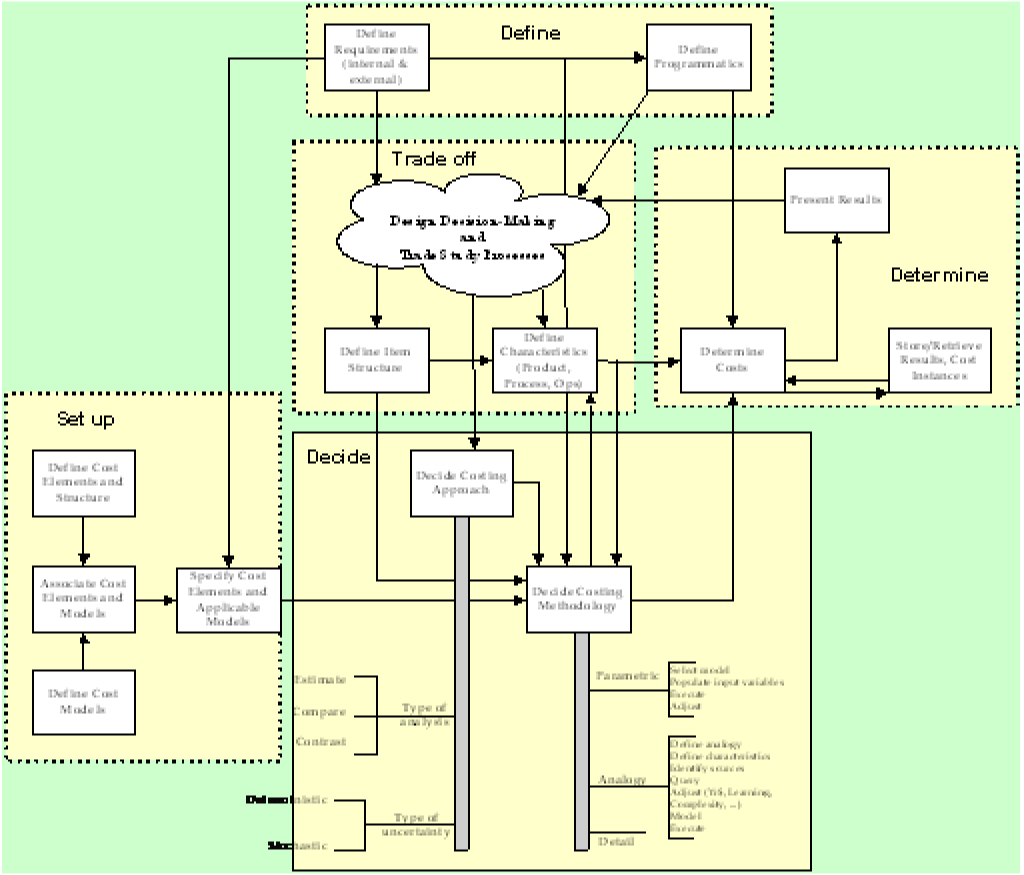
- create
- query
- modify
- store
- retrieve
- report

Mississippi State University
February 2003

FINAL REPORT: Pathfinder 2: Insitu Design Cost Trades (IDCT) Tool Contract: F33615-00-C-5902

7

Cost analysis relationship map

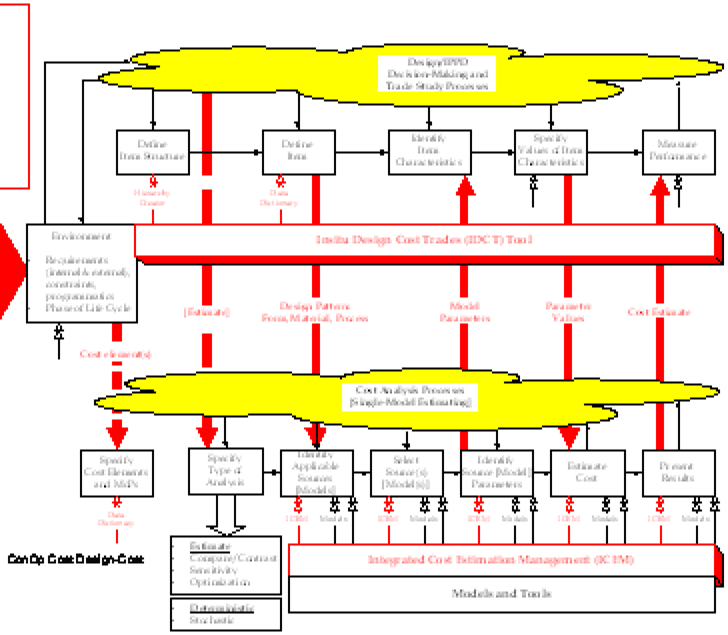


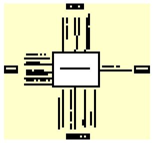
Understanding design, requirements, and cost analysis processes are essential to providing an integrated decision-support tool.

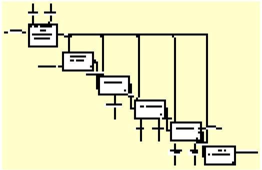
In order to effectively *design-in affordability* early in design, IPTs need tools that are:
- integrated into the design process,
- interact with tools supporting other types of analyses

Insertion of these technologies require an understanding of the design, requirements, and cost-estimation processes.

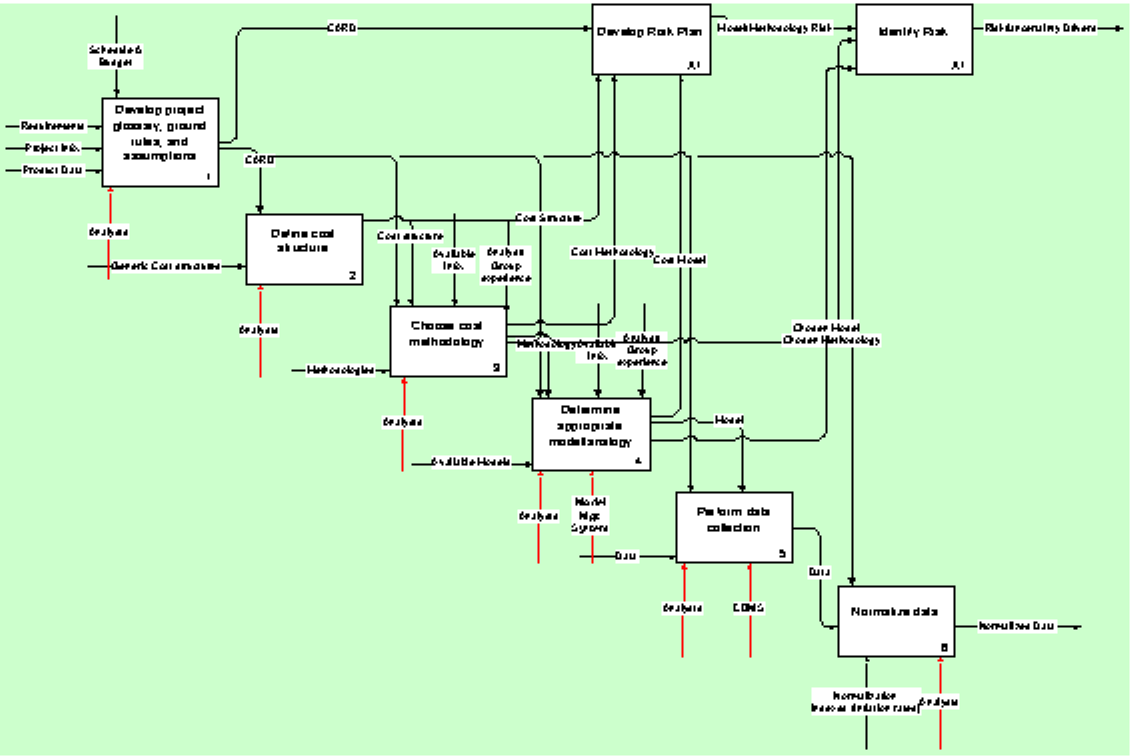
Defining the primary processes identifies:
- activities that need to be supported,
- tools that are used in those activities that must be integrated, and
- means to insert tools into the process



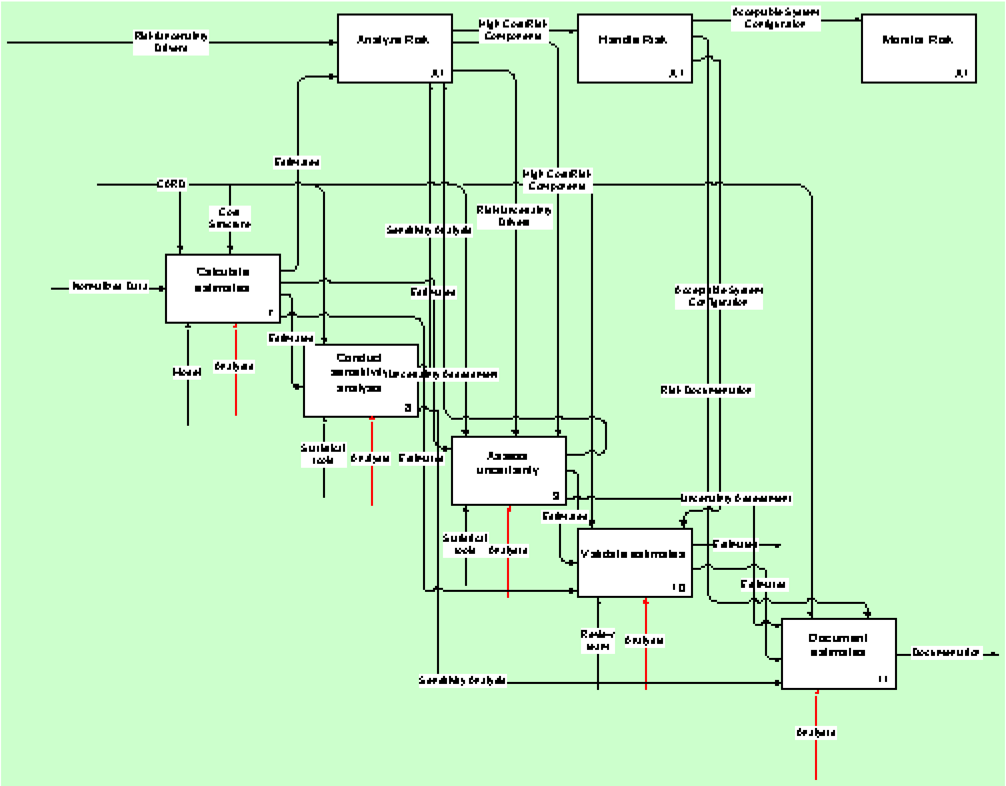




IDEF$_0$ representation of a generic cost estimation process, as derived from the literature.

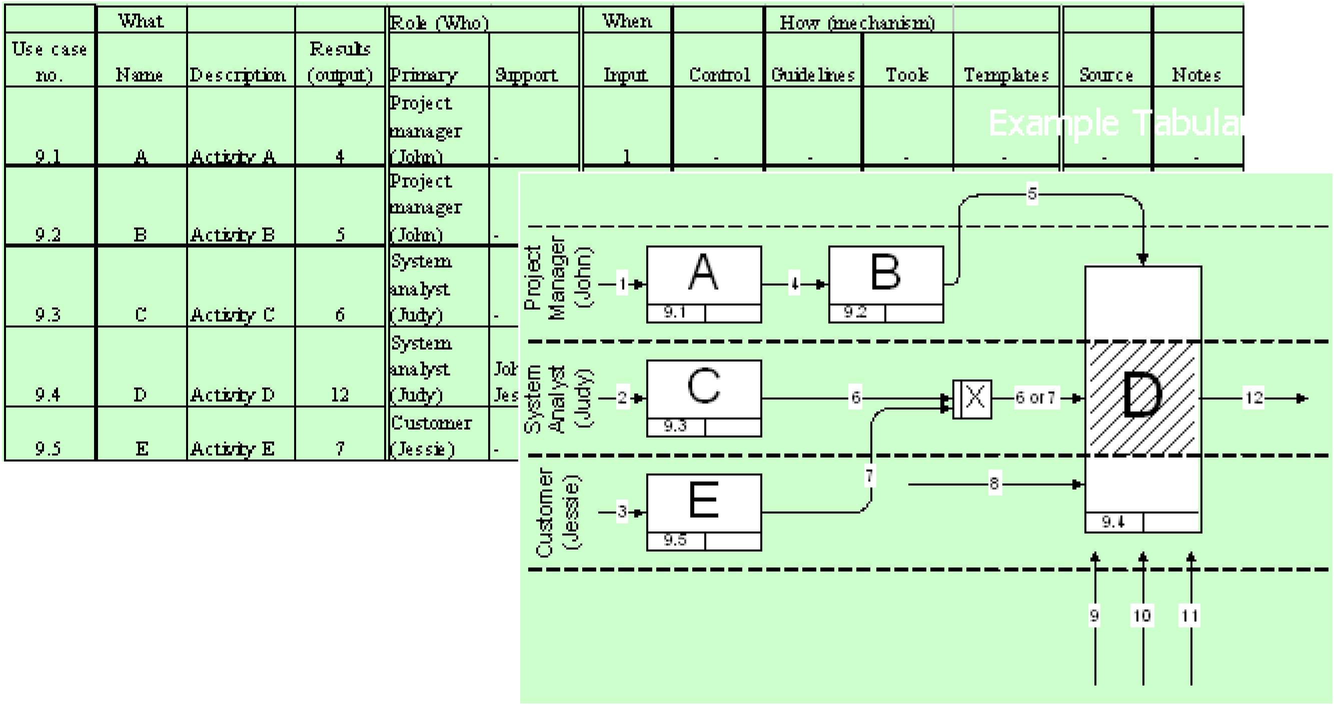


IDEF$_0$ representation of a generic cost estimation process, as derived from the literature (cont.)



Requirements process

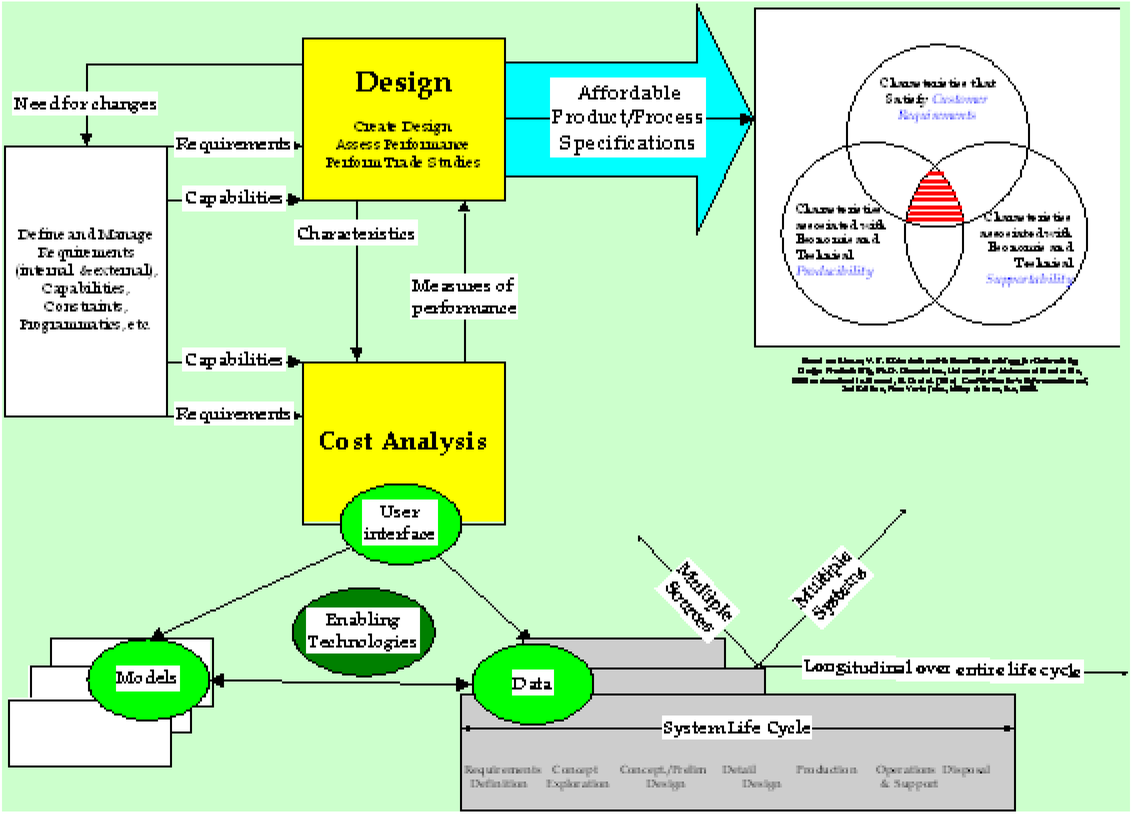
Representation methodologies that were developed



	What			Role (Who)		When		How (mechanism)				
Use case no.	Name	Description	Results (output)	Primary	Support	Input	Control	Guidelines	Tools	Templates	Source	Notes
9.1	A	Activity A	4	Project manager (John)	-	1	-	-	-	-	-	-
9.2	B	Activity B	5	Project manager (John)	-							
9.3	C	Activity C	6	System analyst (Judy)	-							
9.4	D	Activity D	12	System analyst (Judy)	Joh Jes							
9.5	E	Activity E	7	Customer (Jessie)	-							

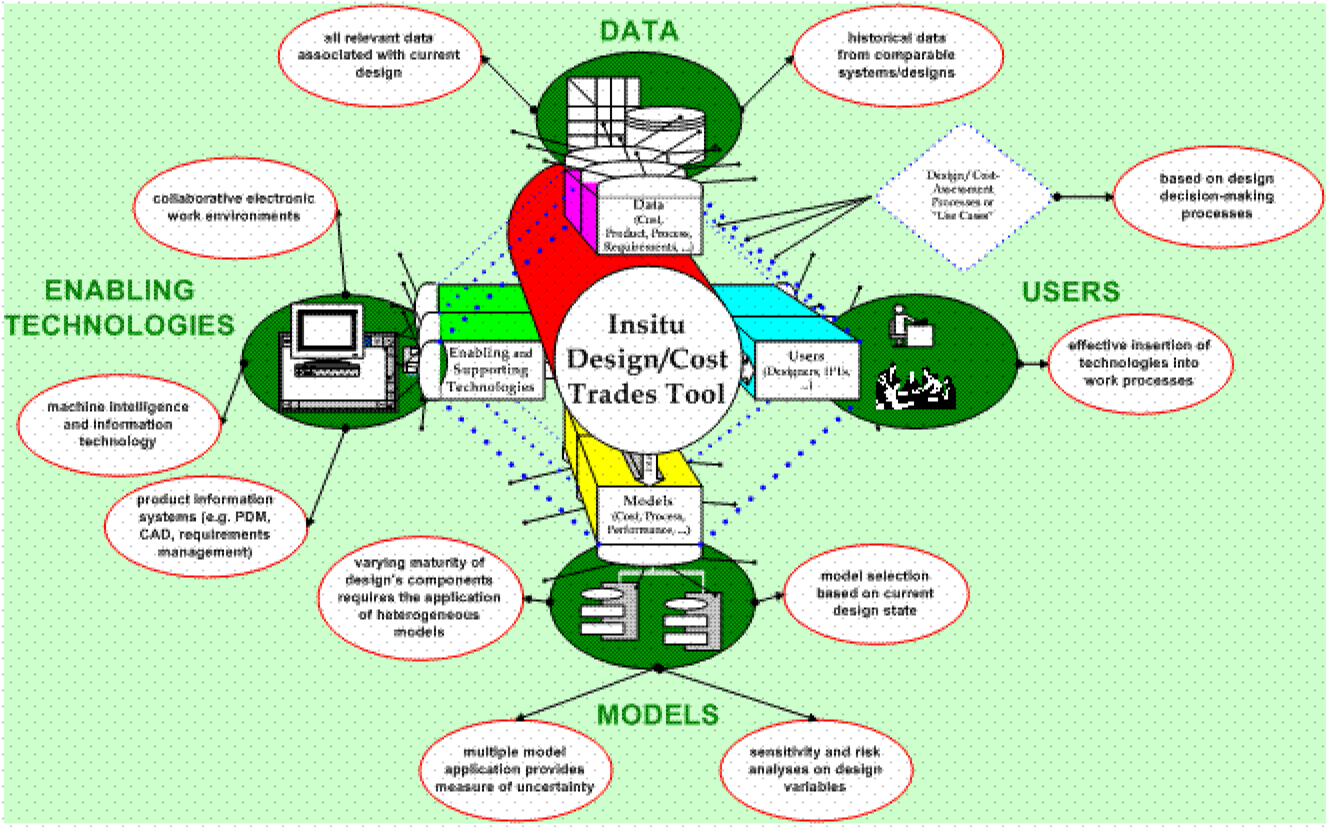
Example Tabular

The design of affordable products require tight links to cost and risk analyses



[illegible]

Decision support approach & example issues



DATA

all relevant data associated with current design

historical data from comparable systems/designs

Design/ Cost-Assessment Processes or "Use Cases"

based on design decision-making processes

ENABLING TECHNOLOGIES

collaborative electronic work environments

machine intelligence and information technology

product information systems (e.g. PDM, CAD, requirements management)

USERS

effective insertion of technologies into work processes

Data (Cost, Product, Process, Requirements, ...)

Enabling and Supporting Technologies

Insitu Design/Cost Trades Tool

Users (Designers, IPTs, ...)

Models (Cost, Process, Performance, ...)

varying maturity of design's components requires the application of heterogeneous models

model selection based on current design state

MODELS

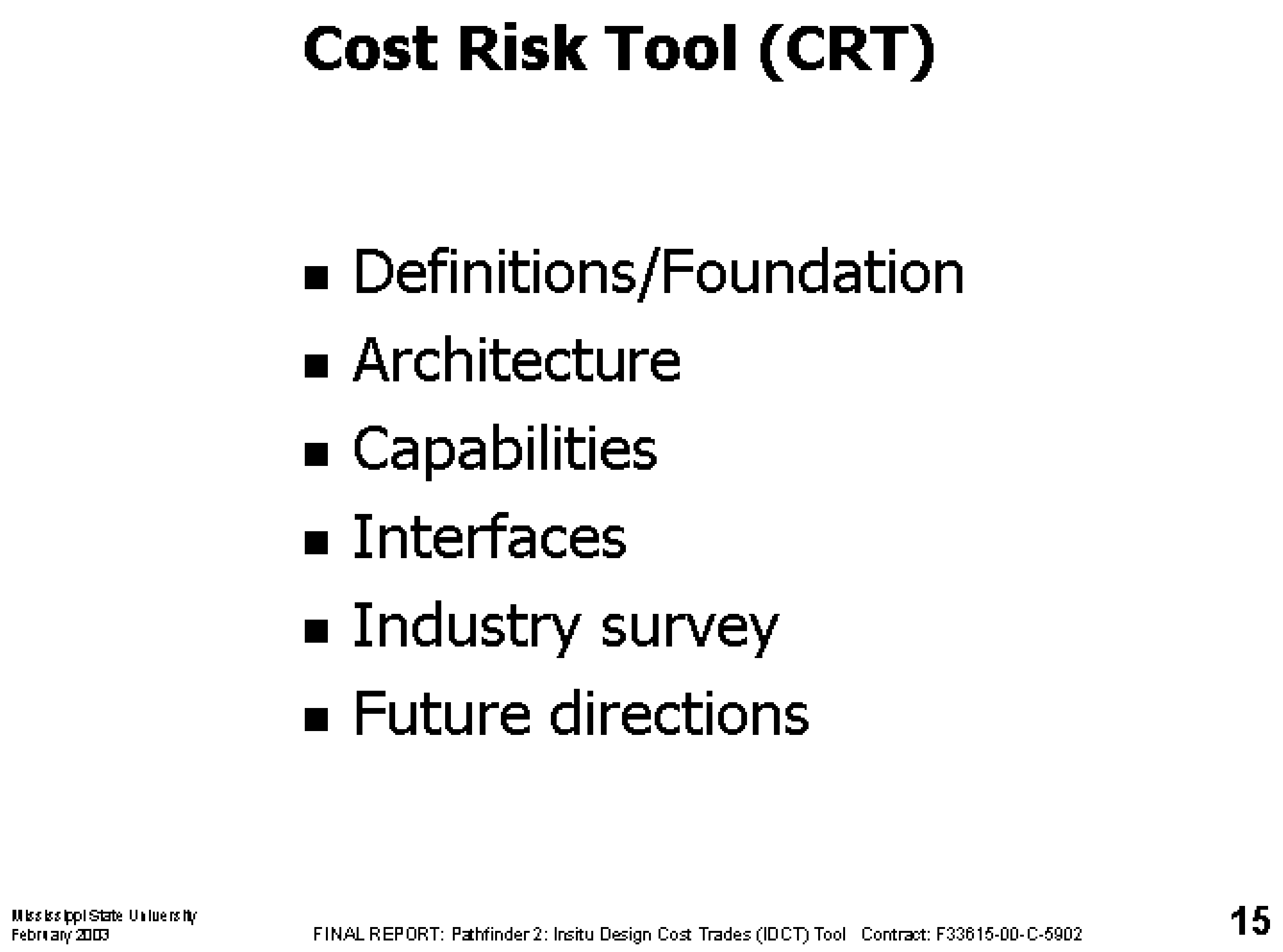
multiple model application provides measure of uncertainty

sensitivity and risk analyses on design variables

Mississippi State University
February 2003

FINAL REPORT: Pathfinder 2: Insitu Design Cost Trades (IDCT) Tool   Contract: F33615-00-C-5902

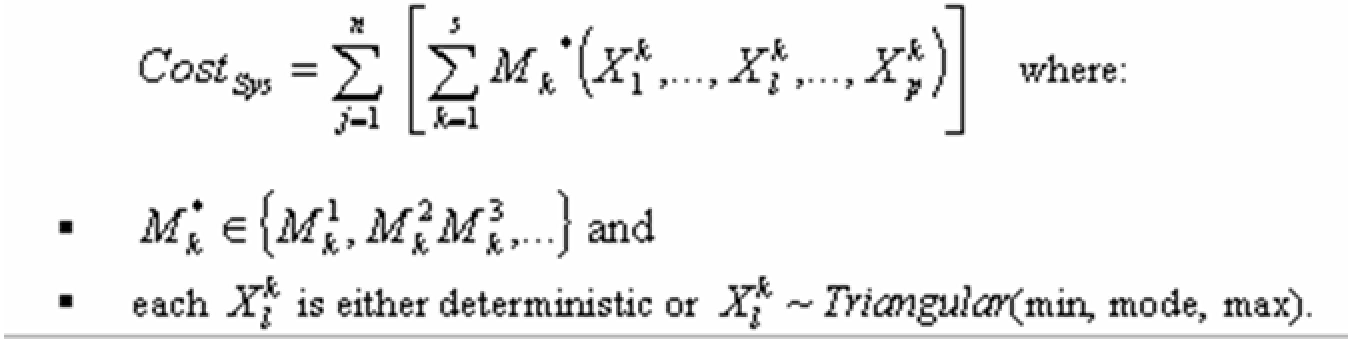
14



Cost Risk Tool (CRT)

- Definitions/Foundation
- Architecture
- Capabilities
- Interfaces
- Industry survey
- Future directions

Definitions and Assertions

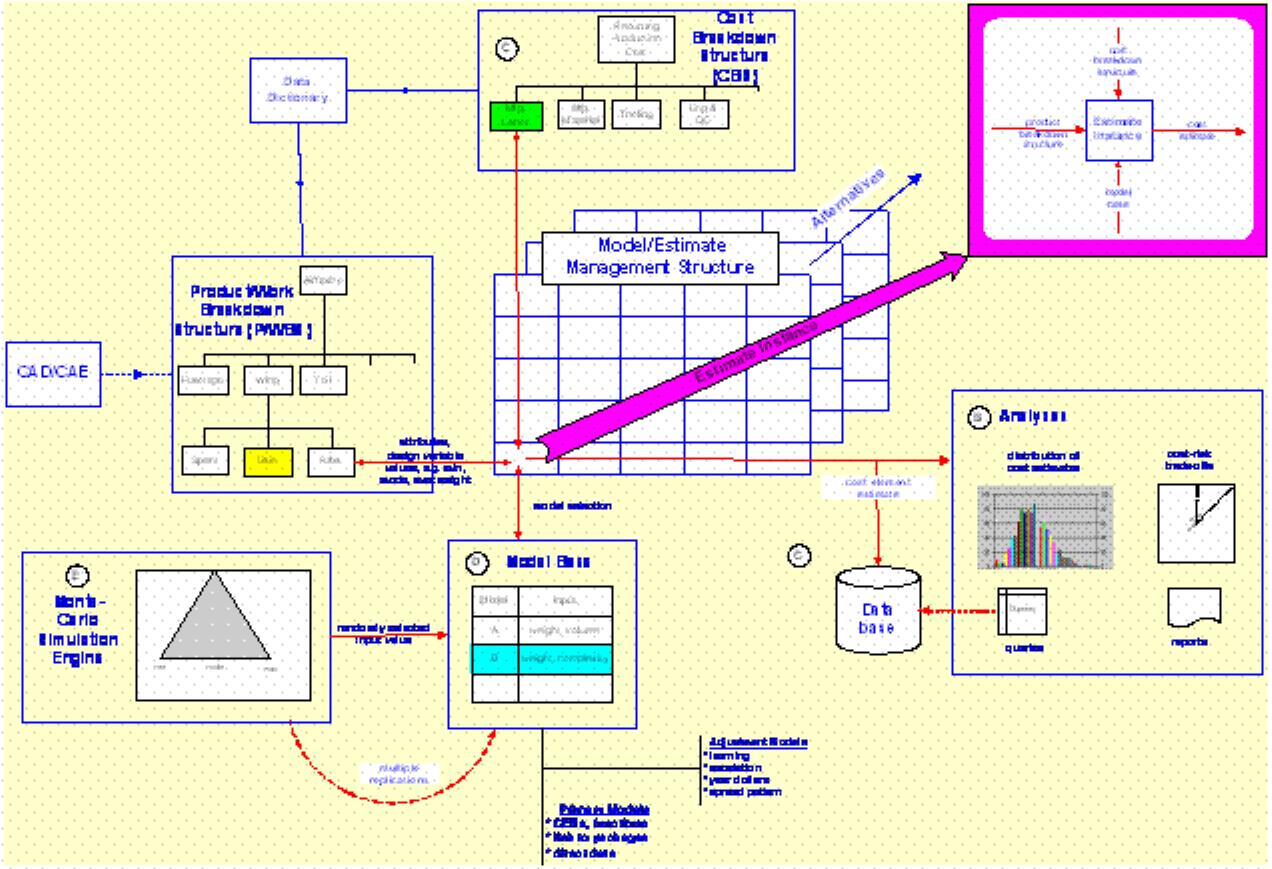


$$Cost_{Sys} = \sum_{j=1}^{n}\left[\sum_{k=1}^{s} M_k^{\bullet}\left(X_1^k,\ldots,X_l^k,\ldots,X_p^k\right)\right] \quad \text{where:}$$

- $M_k^{\bullet} \in \left\{M_k^1, M_k^2 M_k^3,\ldots\right\}$ and
- each X_l^k is either deterministic or $X_l^k \sim \mathit{Triangular}(\text{min, mode, max})$.

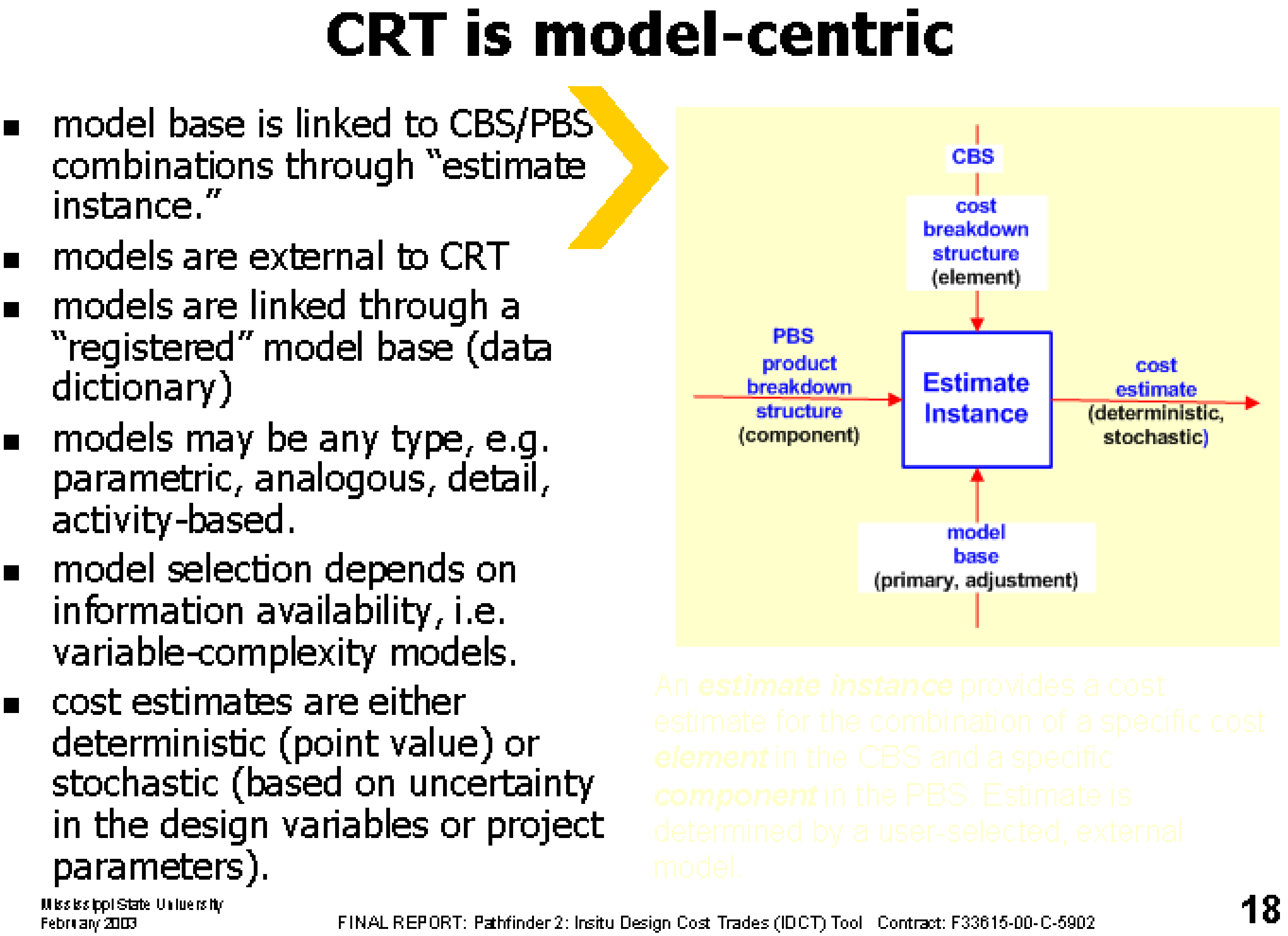
- a system is composed of *n components*
- the system and its components are characterized by *p* variables/parameters, X_l
- “cost” is composed of *s elements*
- one of many potential models, M_k, is used to estimate cost element *k*, based on the characteristics of the system and/or component *j* (variables/parameters, X_l)

Overall CRT architecture



Mississippi State University
February 2003

FINAL REPORT: Pathfinder 2: Insitu Design Cost Trades (IDCT) Tool Contract: F33615-00-C-5902



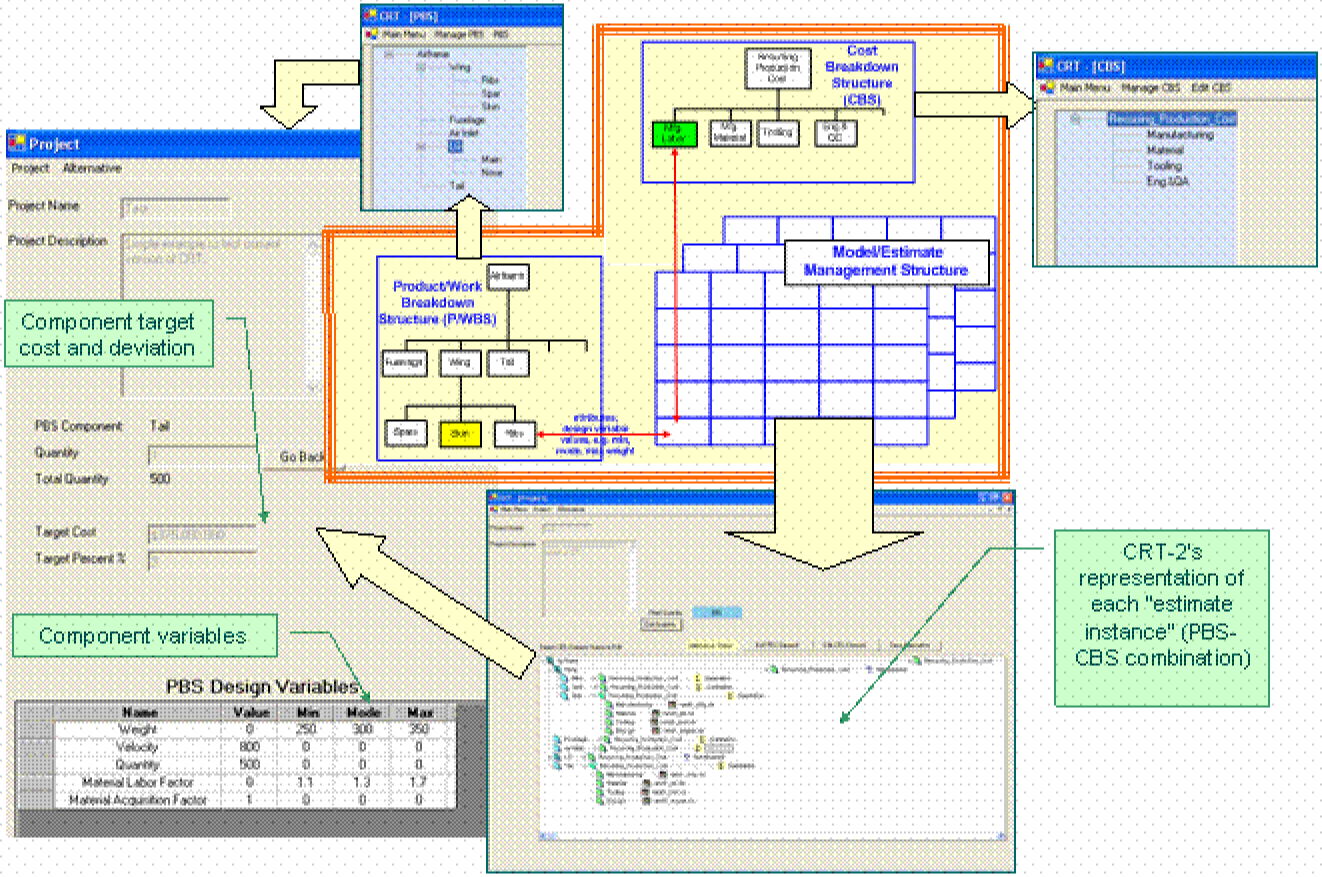
CRT is model-centric

- model base is linked to CBS/PBS combinations through "estimate instance."
- models are external to CRT
- models are linked through a "registered" model base (data dictionary)
- models may be any type, e.g. parametric, analogous, detail, activity-based.
- model selection depends on information availability, i.e. variable-complexity models.
- cost estimates are either deterministic (point value) or stochastic (based on uncertainty in the design variables or project parameters).

CBS
cost breakdown structure (element)
PBS
product breakdown structure (component)
Estimate Instance
cost estimate (deterministic, stochastic)
model base (primary, adjustment)

An ***estimate instance*** provides a cost estimate for the combination of a specific cost ***element*** in the CBS and a specific ***component*** in the PBS. Estimate is determined by a user-selected, external model.

Defining the product and cost hierarchies



PBS Component	Tail
Quantity	1
Total Quantity	500
Target Cost	
Target Percent %	

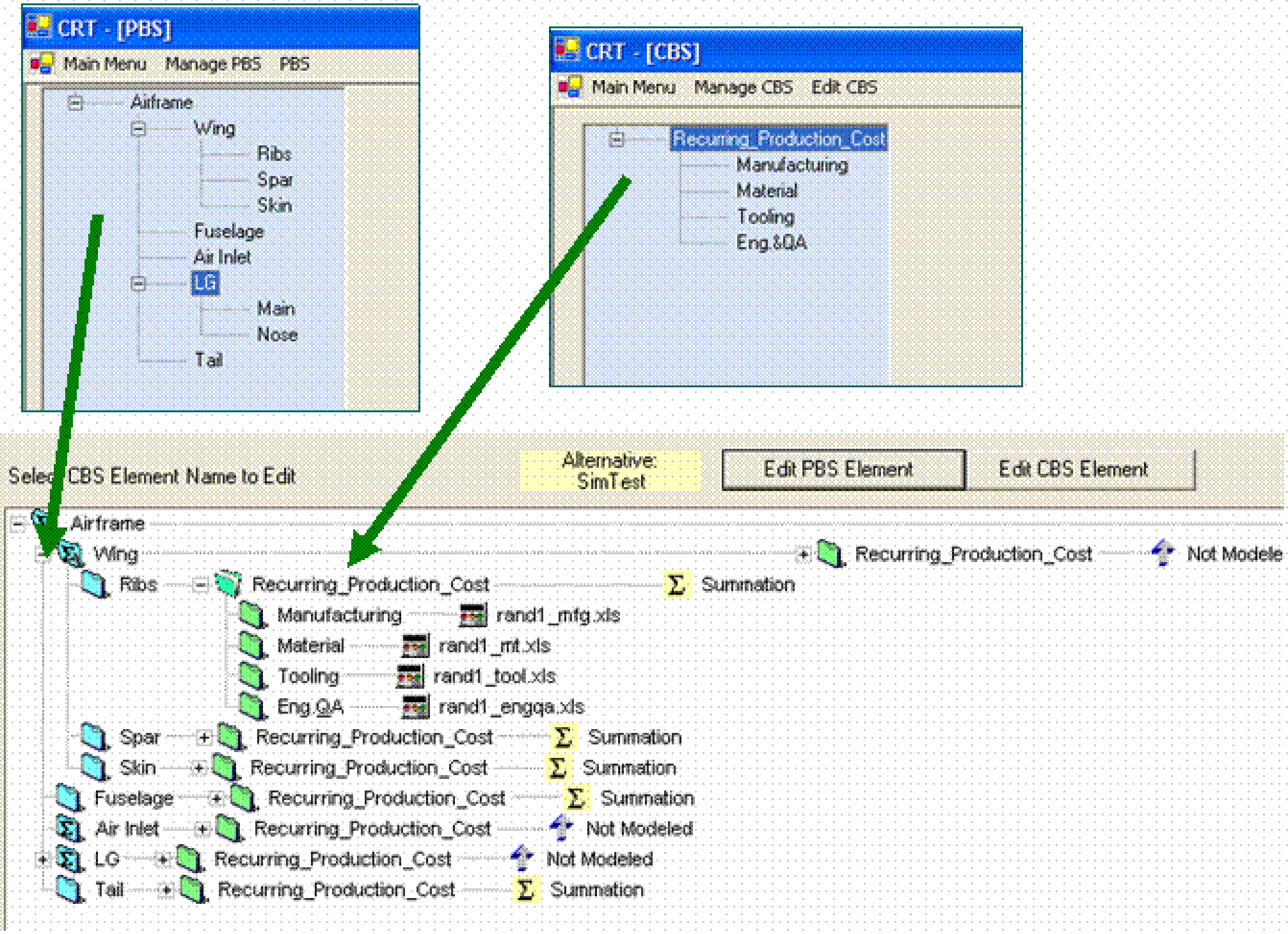
Component target cost and deviation

Component variables

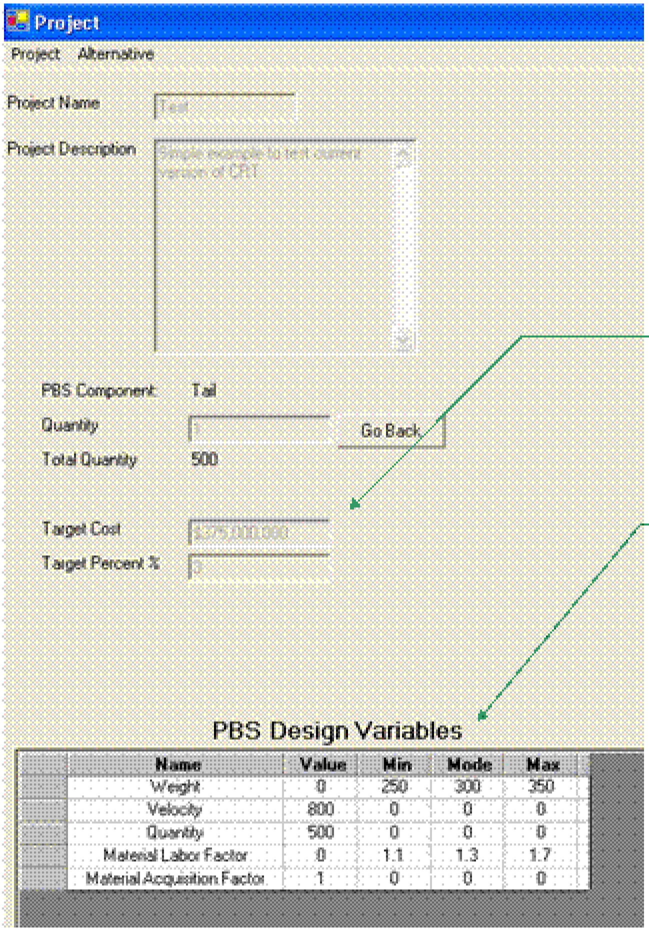
CRT-2's representation of each "estimate instance" (PBS-CBS combination)

PBS Design Variables

Name	Value	Min	Mode	Max
Weight	0	250	300	350
Velocity	800	0	0	0
Quantity	500	0	0	0
Material Labor Factor	0	1.1	1.3	1.7
Material Acquisition Factor	1	0	0	0


CRT - [PBS]
Main Menu
Manage PBS
PBS
Airframe
Wing
Ribs
Spar
Skin
Fuselage
Air Inlet
LG
Main
Nose
Tail
CRT - [CBS]
Main Menu
Manage CBS
Edit CBS
Recurring_Production_Cost
Manufacturing
Material
Tooling
Eng.&QA
Select CBS Element Name to Edit
Alternative: SimTest
Edit PBS Element
Edit CBS Element
Airframe
Wing
Recurring_Production_Cost
Not Modele
Ribs
Recurring_Production_Cost
Summation
Manufacturing
rand1_mfg.xls
Material
rand1_mt.xls
Tooling
rand1_tool.xls
Eng.QA
rand1_engqa.xls
Spar
Recurring_Production_Cost
Summation
Skin
Recurring_Production_Cost
Summation
Fuselage
Recurring_Production_Cost
Summation
Air Inlet
Recurring_Production_Cost
Not Modeled
LG
Recurring_Production_Cost
Not Modeled
Tail
Recurring_Production_Cost
Summation

PBS component information



Project Name: Test

Project Description: Simple example to test current version of CRT

PBS Component: Tail

Quantity: 1

Total Quantity: 500

Target Cost: $275,000,000

Target Percent %: 0

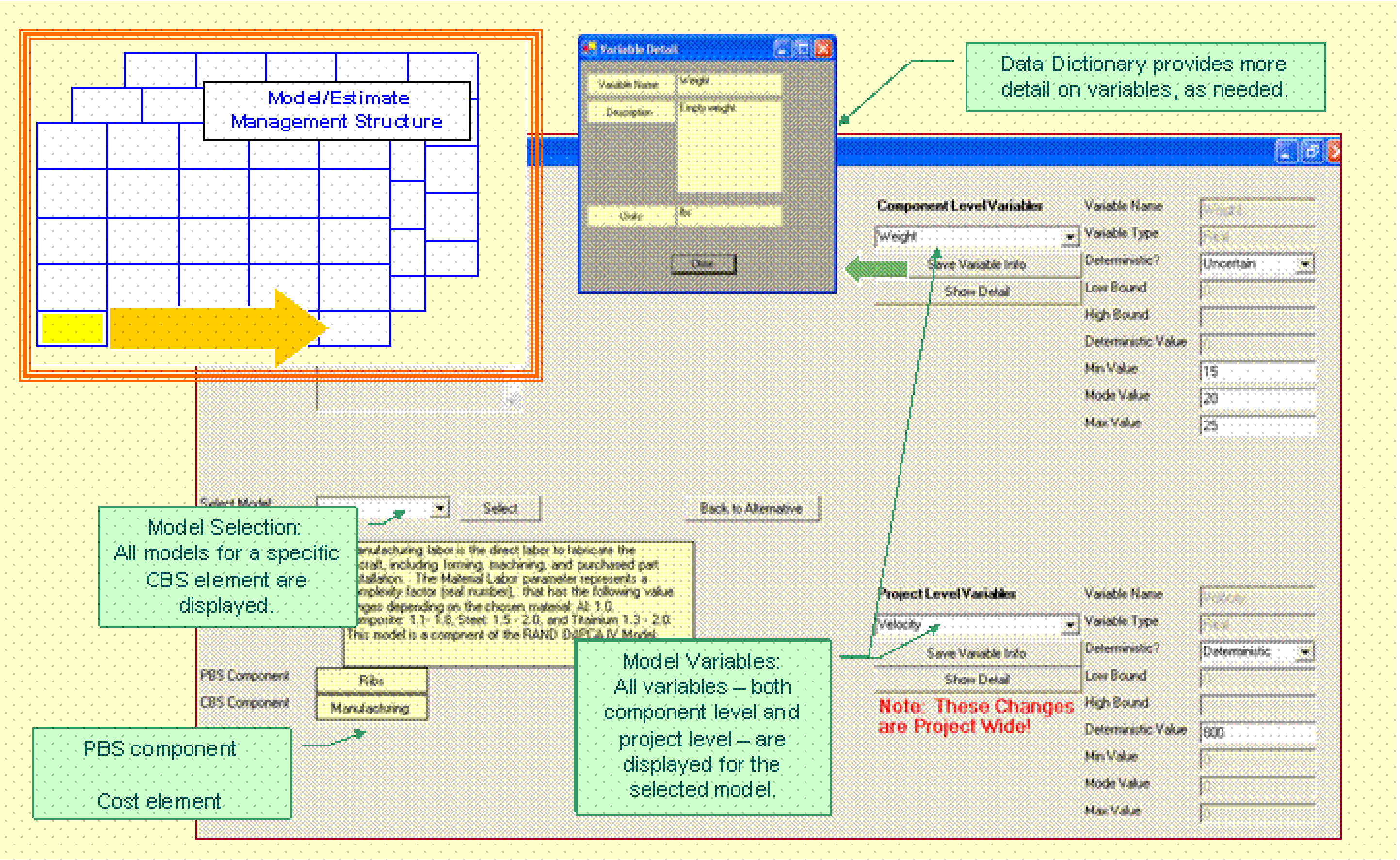
PBS Design Variables

Name	Value	Min	Mode	Max
Weight	0	250	300	350
Velocity	800	0	0	0
Quantity	500	0	0	0
Material Labor Factor	0	1.1	1.3	1.7
Material Acquisition Factor	1	0	0	0

Component target cost and deviation

Component variables

Model selection and variable definition



Variable Detail

Variable Name	Weight
Description	Empty weight
Units	lbs

Data Dictionary provides more detail on variables, as needed.

Component Level Variables

Weight

Variable Name	Weight
Variable Type	Real
Deterministic?	Uncertain
Low Bound	0
High Bound	
Deterministic Value	0
Min Value	15
Mode Value	20
Max Value	25

Select Model

anufacturing labor is the direct labor to fabricate the craft, including forming, machining, and purchased part stallation. The Material Labor parameter represents a mplexity factor (real number), that has the following value nges depending on the chosen material: Al: 1.0, mposite: 1.1- 1.8, Steel: 1.5 - 2.0, and Titanium 1.3 - 2.0. This model is a component of the RAND DAPCA IV Model.

Model Selection:
All models for a specific CBS element are displayed.

PBS Component	Ribs
CBS Component	Manufacturing

PBS component

Cost element

Model Variables:
All variables – both component level and project level – are displayed for the selected model.

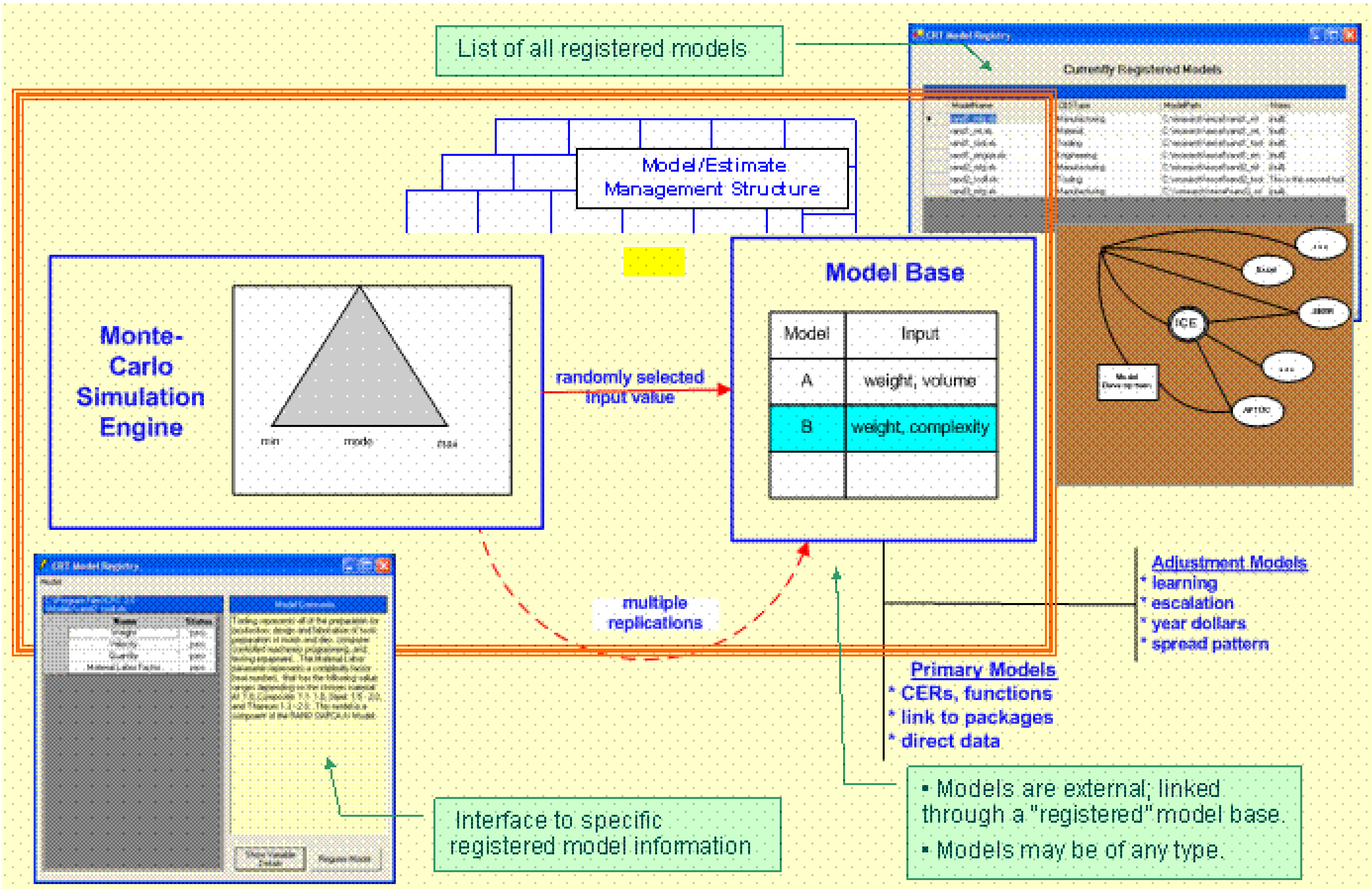
Project Level Variables

Velocity

Note: These Changes are Project Wide!

Variable Name	Velocity
Variable Type	Real
Deterministic?	Deterministic
Low Bound	0
High Bound	
Deterministic Value	800
Min Value	0
Mode Value	0
Max Value	0

Model base


Model/Estimate Management Structure
Monte-Carlo Simulation Engine
min
mode
max
randomly selected input value
multiple replications
Model Base
Currently Registered Models
ICE
Excel
Model Development

List of all registered models

Model	Input
A	weight, volume
B	weight, complexity

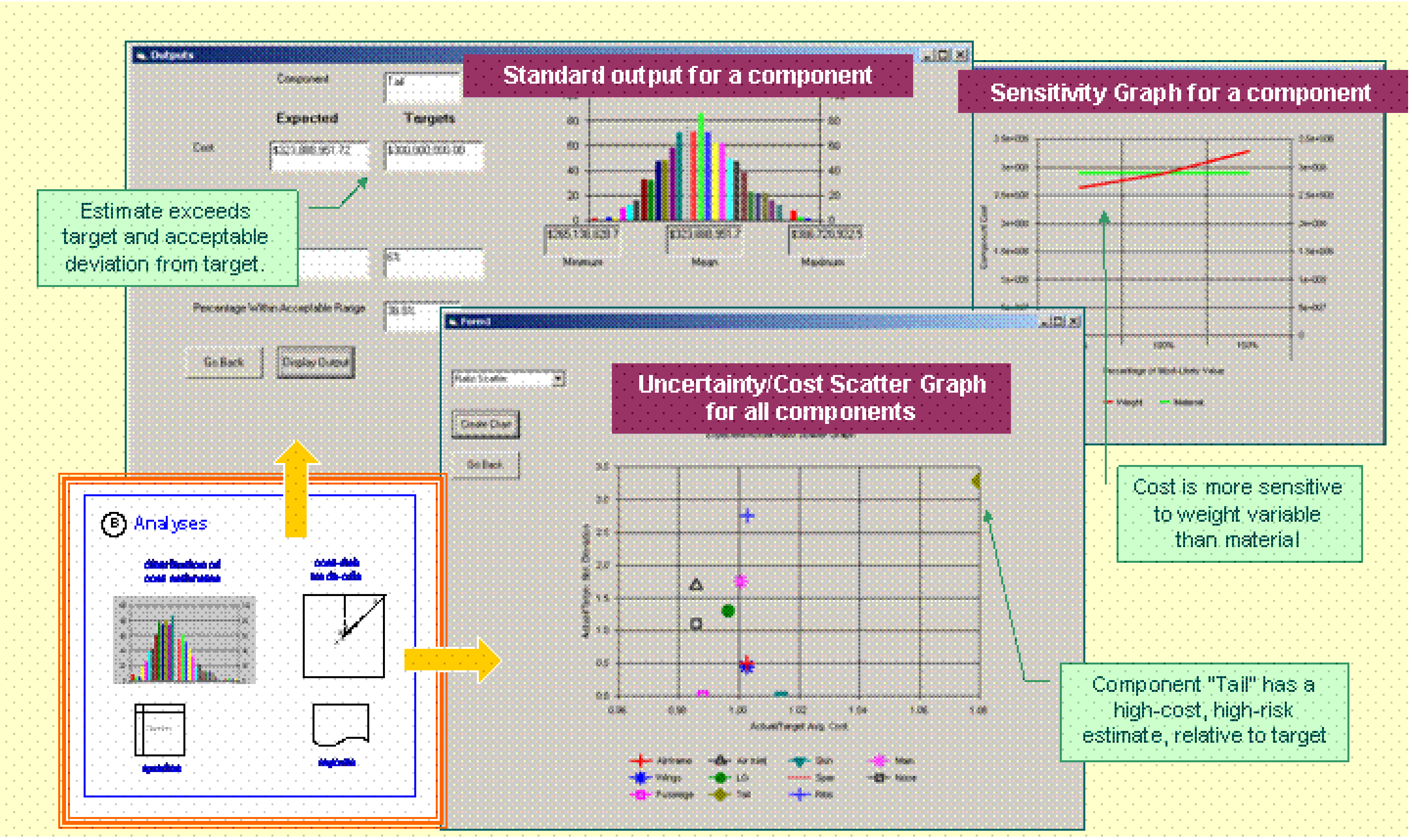
Adjustment Models
- learning
- escalation
- year dollars
- spread pattern

Primary Models
- CERs, functions
- link to packages
- direct data

Interface to specific registered model information

- Models are external; linked through a "registered" model base.
- Models may be of any type.

Output – cost/risk analysis support



Standard output for a component

Sensitivity Graph for a component

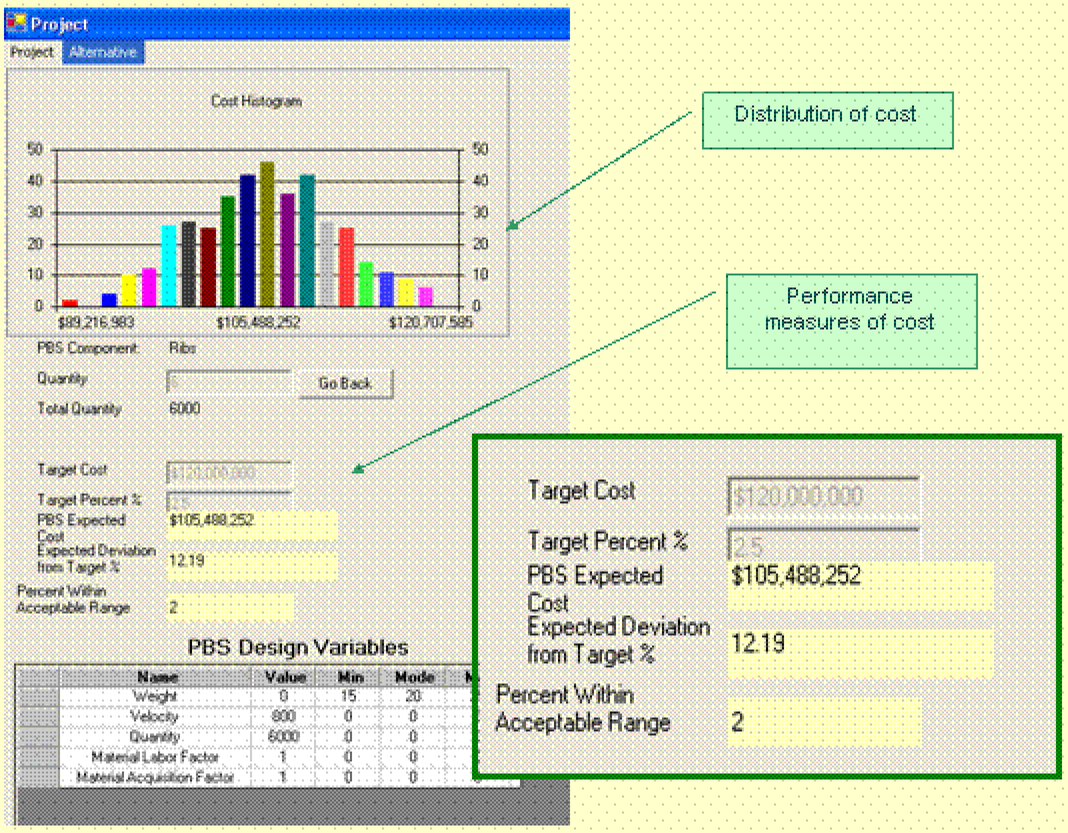
Estimate exceeds target and acceptable deviation from target.

Uncertainty/Cost Scatter Graph for all components

Cost is more sensitive to weight variable than material

Component "Tail" has a high-cost, high-risk estimate, relative to target

Component analysis support



Distribution of cost

Performance measures of cost

PBS Component Ribs

Quantity 6

Go Back

Total Quantity 6000

Target Cost $120,000,000

Target Percent % 2.5

PBS Expected Cost $105,488,252

Expected Deviation from Target % 12.19

Percent Within Acceptable Range 2

PBS Design Variables

Name	Value	Min	Mode
Weight	0	15	20
Velocity	800	0	0
Quantity	6000	0	0
Material Labor Factor	1	0	0
Material Acquisition Factor	1	0	0

Target Cost $120,000,000

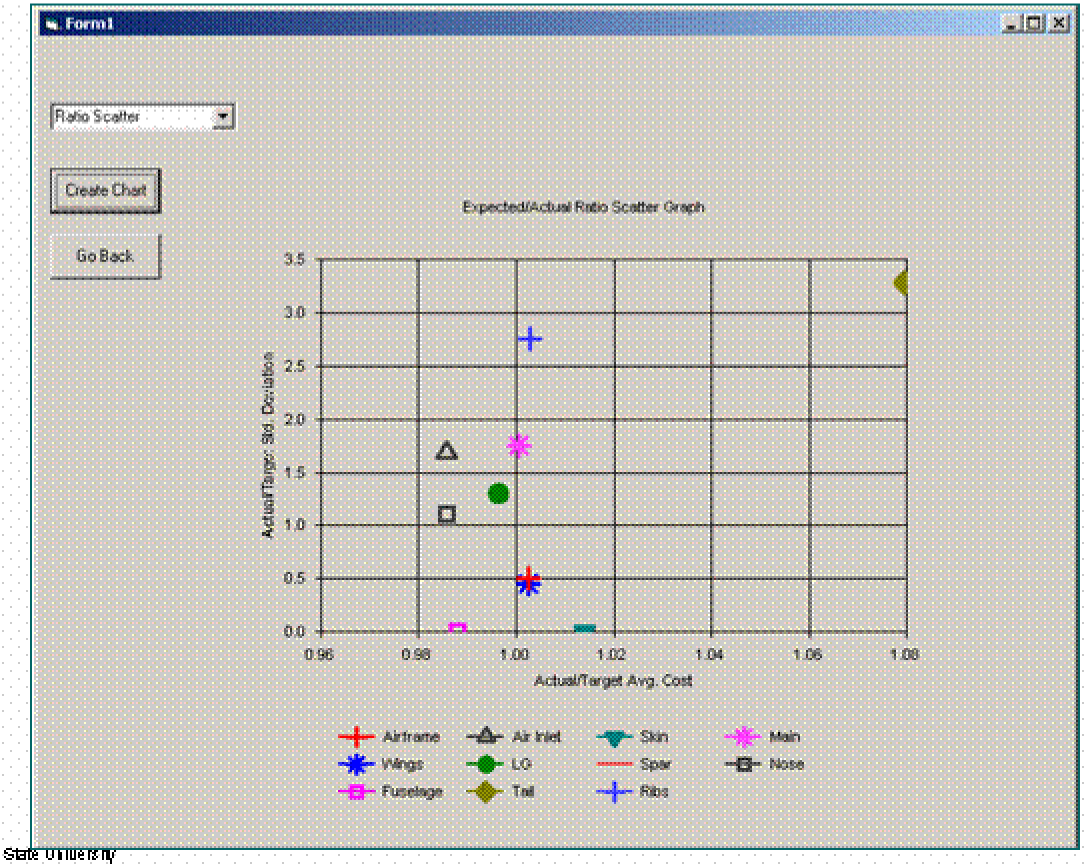
Target Percent % 2.5

PBS Expected Cost $105,488,252

Expected Deviation from Target % 12.19

Percent Within Acceptable Range 2

Cost/Risk scatter graph

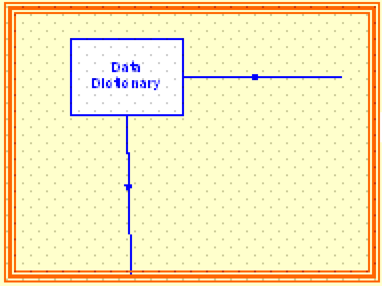


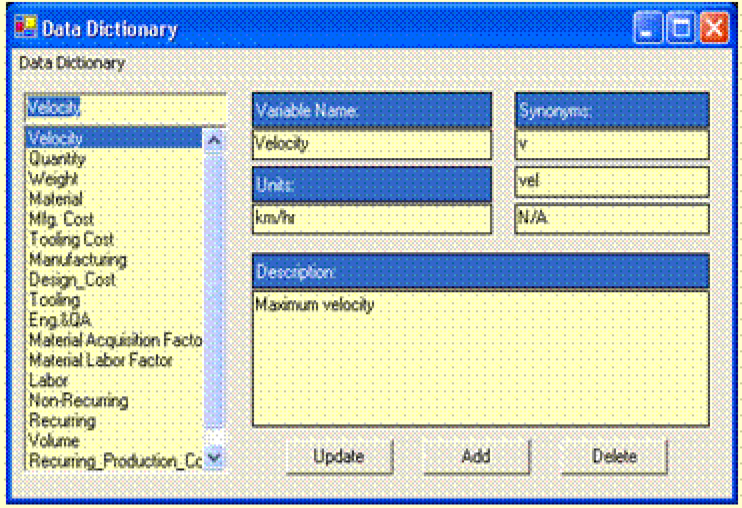
Performance measures

- Mean, standard deviation, and distribution of estimated cost
- Mean deviation between estimated cost and target cost
- Probability that cost will be within target interval
- Cost ratio (expected cost / target cost)
- Risk ratio (expected deviation / acceptable deviation)

Data dictionary

Variables, parameters, and CBS elements are defined in a data dictionary; includes units and synonyms





Data Dictionary

Velocity

Velocity
Quantity
Weight
Material
Mfg. Cost
Tooling Cost
Manufacturing
Design_Cost
Tooling
Eng.&QA
Material Acquisition Facto
Material Labor Factor
Labor
Non-Recurring
Recurring
Volume
Recurring_Production_Cc

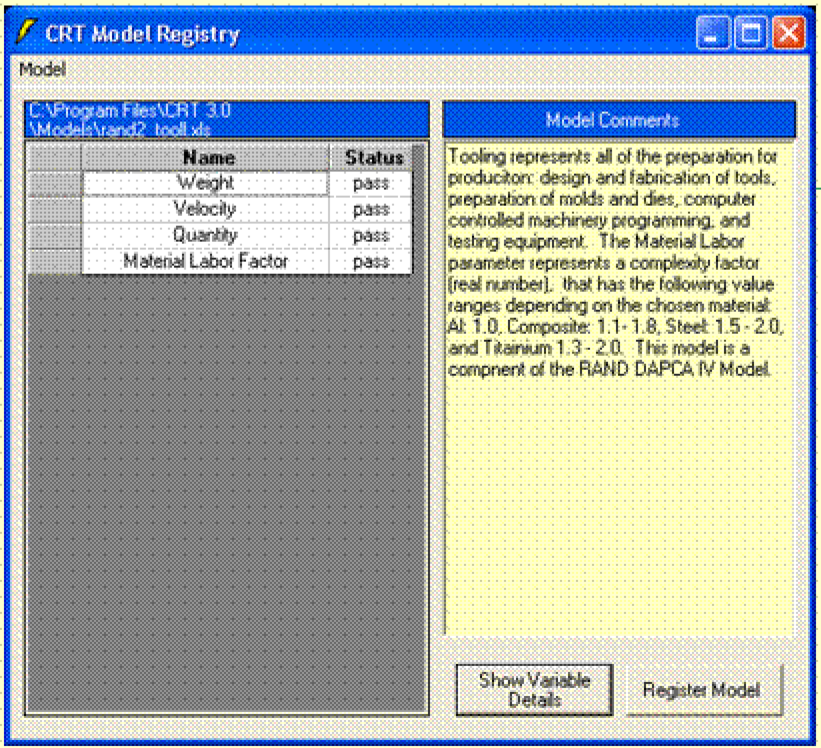
Variable Name: Velocity

Units: km/hr

Synonyms: v, vel, N/A

Description: Maximum velocity

Model registration



Name	Status
Weight	pass
Velocity	pass
Quantity	pass
Material Labor Factor	pass

Tooling represents all of the preparation for producitonː design and fabrication of tools, preparation of molds and dies, computer controlled machinery programming, and testing equipment. The Material Labor parameter represents a complexity factor (real number), that has the following value ranges depending on the chosen material: Al: 1.0, Composite: 1.1- 1.8, Steel: 1.5 - 2.0, and Titainium 1.3 - 2.0. This model is a compnent of the RAND DAPCA IV Model.

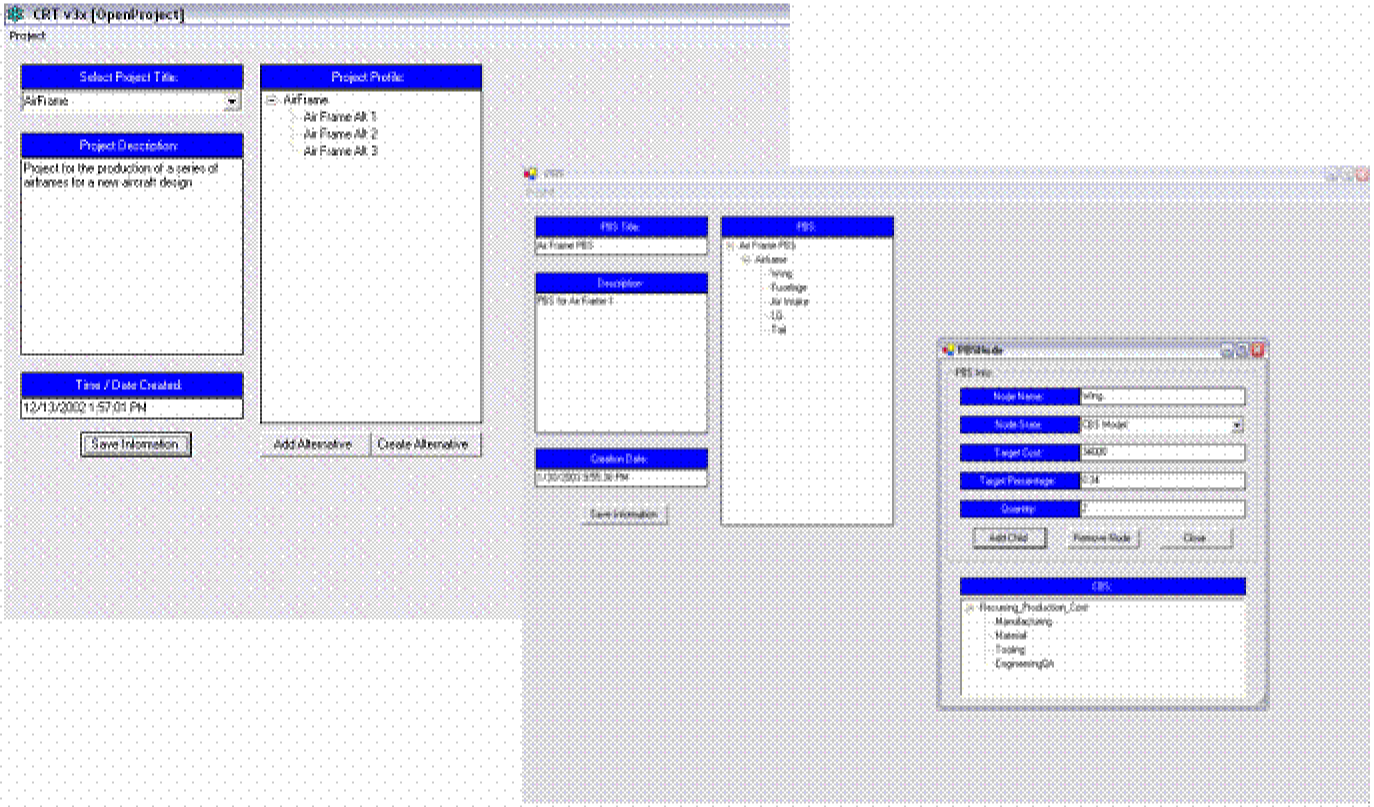
Variables and parameters are defined in a data dictionary; includes units and synonyms

Industry survey

Strongly Disagree	Disagree	Neither agree or disagree	AgreeStrongly	Agree
1	2	3	4	5

- The concepts being developed in this project, as presented in this review, are:
  - important to advancing the cost engineering discipline **4.1**
  - important to enhancing the design of affordable products **4.3**
  - relevant to our business/industry **4.6**
- We support the work being done in this project and encourage the AFRL to continue its funding. **4.4**
- We are willing to provide guidance and other in-kind support for this project's future activities. **4.5**
- We are willing to participate in a test case of the IDCT Tool. **4.3**
- We are willing to jointly fund future development of the project's methodologies/technologies. **2.8**
- We are interested in working with this project to investigate integrating it into our business. **3.9**

New CRT user interfaces



CRT v3x [OpenProject]

Project

Select Project Title: AirFrame

Project Description: Project for the production of a series of airframes for a new aircraft design

Time / Date Created: 12/13/2002 1:57:01 PM

Project Profile:

- AirFrame
  - Air Frame Alt 1
  - Air Frame Alt 2
  - Air Frame Alt 3

PBS

PBS Title: Air Frame PBS

Description: PBS for Air Frame 1

Creation Date: 1/30/2003 3:55:38 PM

PBS:

- Air Frame PBS
  - Airframe
    - Wing
    - Fuselage
    - Air Intake
    - LG
    - Tail

PBSNode

PBS Info

Node Name: Wing

Node State: CBS Model

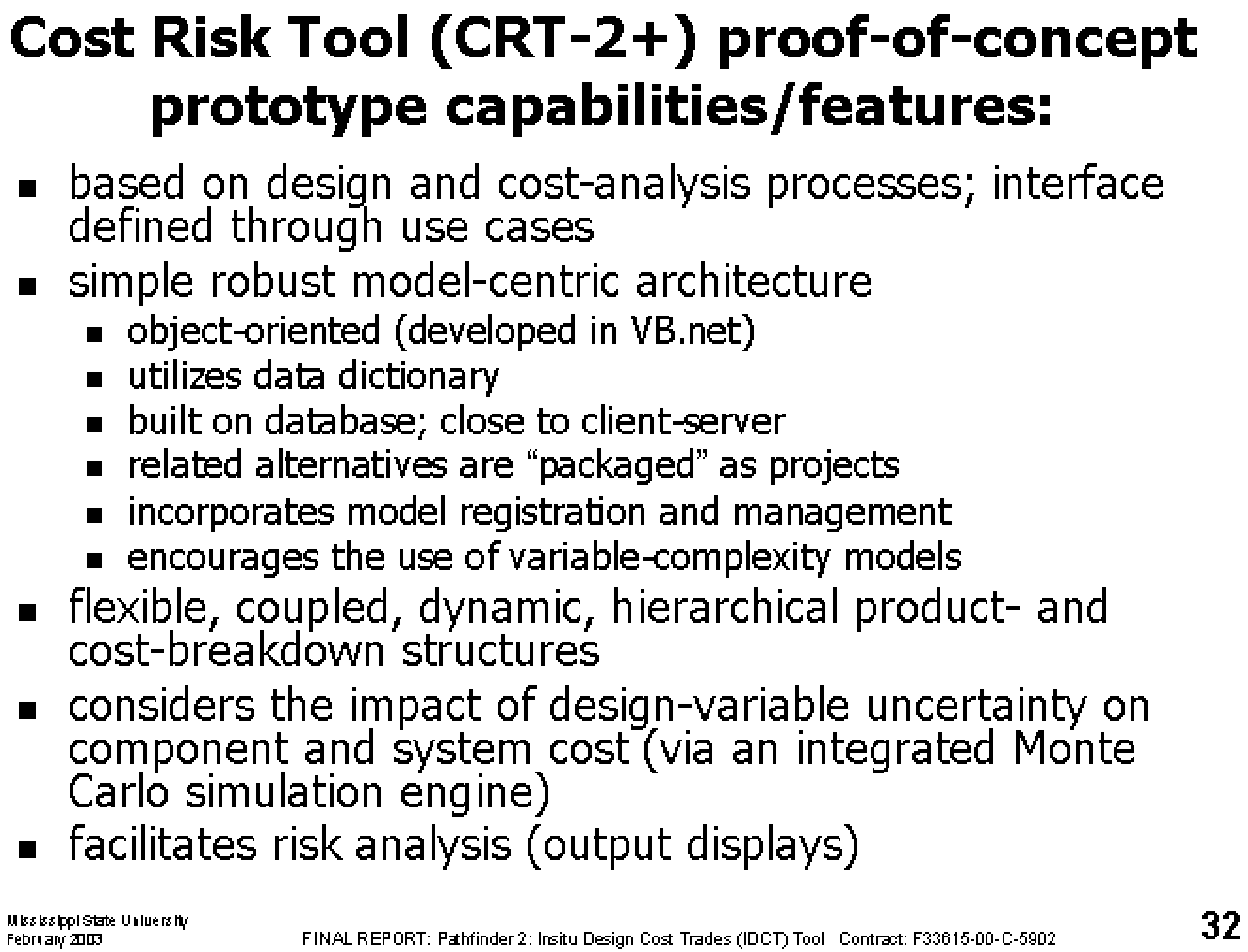
Target Cost: 34000

Target Percentage: 0.34

Quantity: 2

CBS:

- Recurring_Production_Cost
  - Manufacturing
  - Material
  - Tooling
  - EngineeringQA



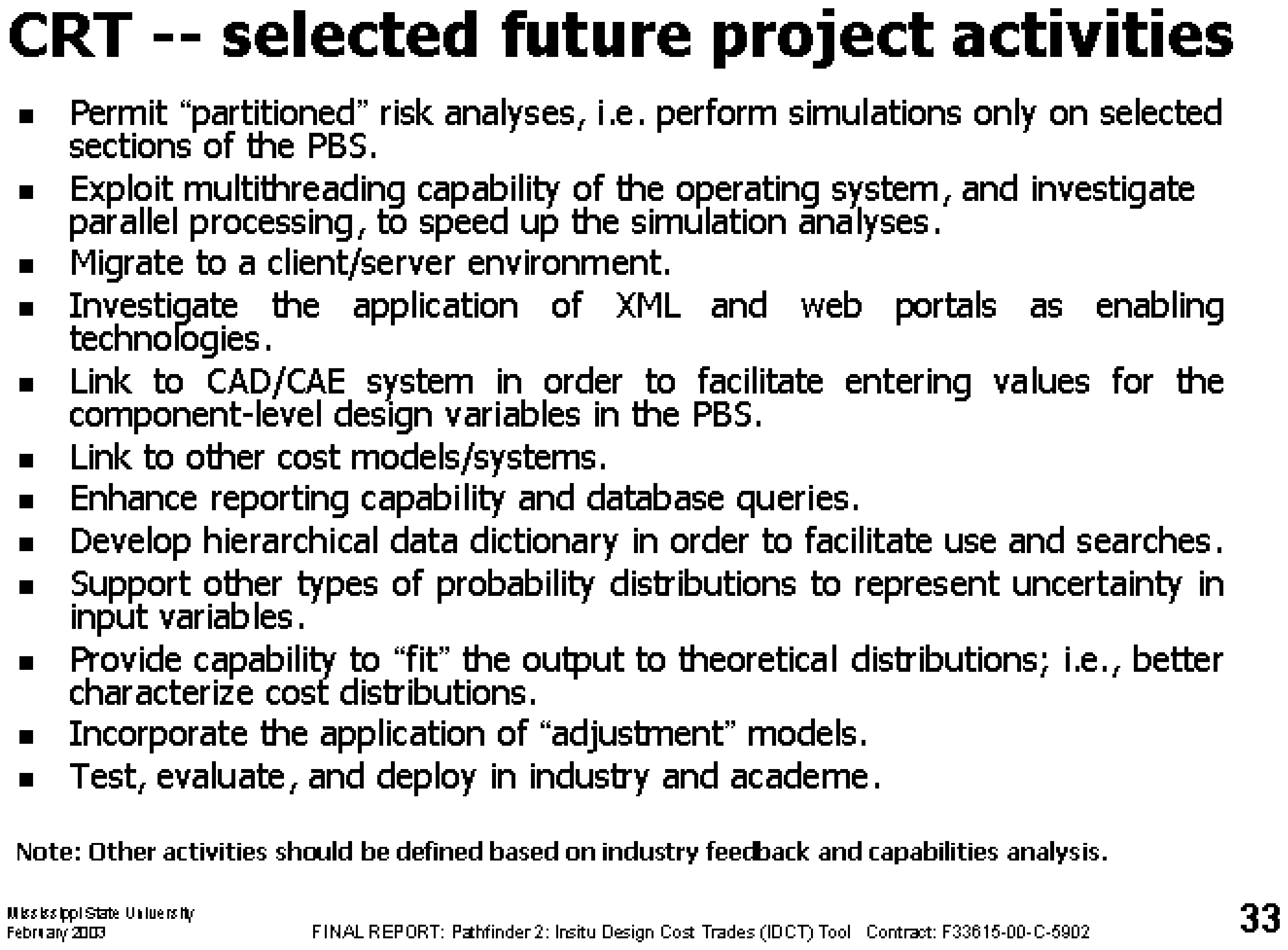
Cost Risk Tool (CRT-2+) proof-of-concept prototype capabilities/features:

- based on design and cost-analysis processes; interface defined through use cases
- simple robust model-centric architecture
  - object-oriented (developed in VB.net)
  - utilizes data dictionary
  - built on database; close to client-server
  - related alternatives are "packaged" as projects
  - incorporates model registration and management
  - encourages the use of variable-complexity models
- flexible, coupled, dynamic, hierarchical product- and cost-breakdown structures
- considers the impact of design-variable uncertainty on component and system cost (via an integrated Monte Carlo simulation engine)
- facilitates risk analysis (output displays)

Mississippi State University
February 2003

FINAL REPORT: Pathfinder 2: Insitu Design Cost Trades (IDCT) Tool Contract: F33615-00-C-5902

32

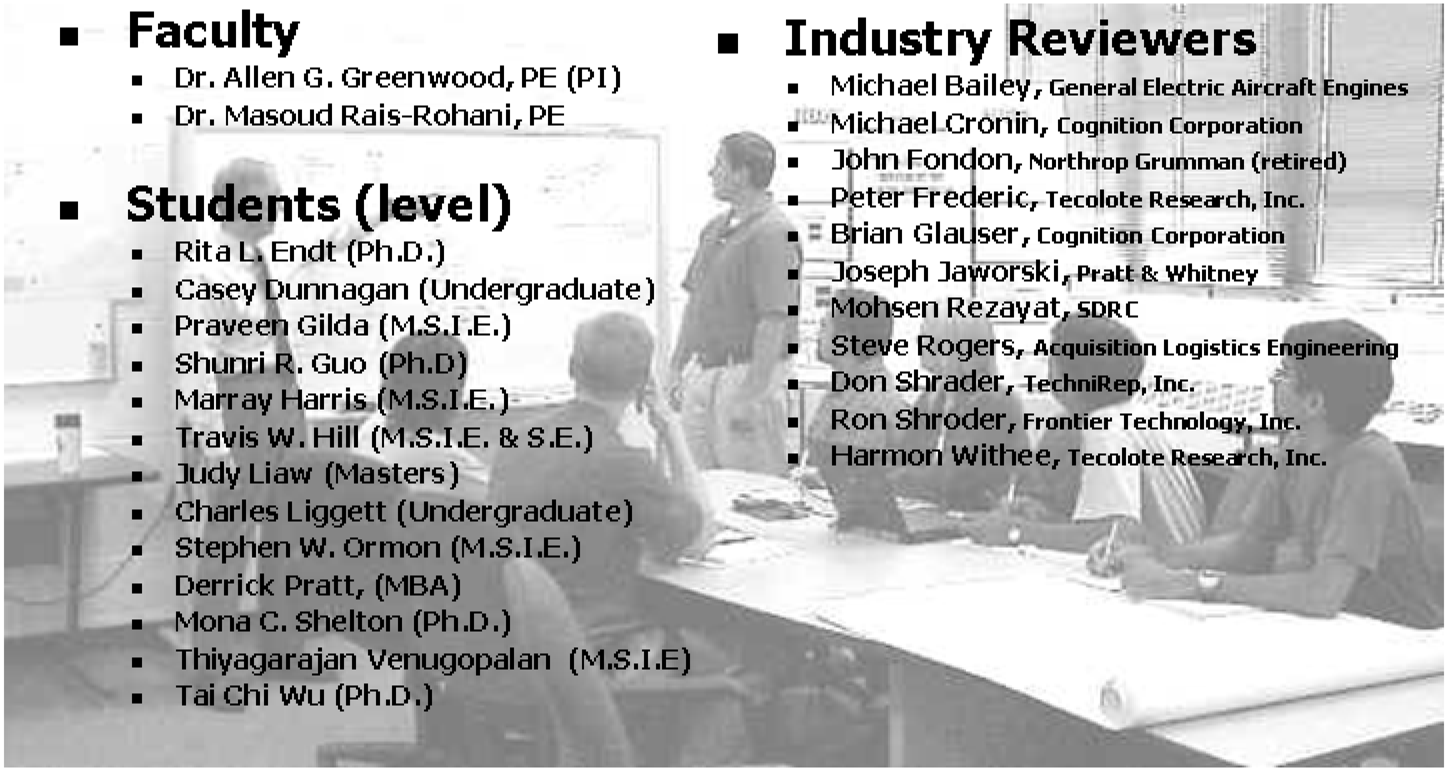


CRT -- selected future project activities

- Permit “partitioned” risk analyses, i.e. perform simulations only on selected sections of the PBS.
- Exploit multithreading capability of the operating system, and investigate parallel processing, to speed up the simulation analyses.
- Migrate to a client/server environment.
- Investigate the application of XML and web portals as enabling technologies.
- Link to CAD/CAE system in order to facilitate entering values for the component-level design variables in the PBS.
- Link to other cost models/systems.
- Enhance reporting capability and database queries.
- Develop hierarchical data dictionary in order to facilitate use and searches.
- Support other types of probability distributions to represent uncertainty in input variables.
- Provide capability to “fit” the output to theoretical distributions; i.e., better characterize cost distributions.
- Incorporate the application of “adjustment” models.
- Test, evaluate, and deploy in industry and academe.

Note: Other activities should be defined based on industry feedback and capabilities analysis.

Participants



- **Faculty**
  - Dr. Allen G. Greenwood, PE (PI)
  - Dr. Masoud Rais-Rohani, PE
- **Students (level)**
  - Rita L. Endt (Ph.D.)
  - Casey Dunnagan (Undergraduate)
  - Praveen Gilda (M.S.I.E.)
  - Shunri R. Guo (Ph.D)
  - Marray Harris (M.S.I.E.)
  - Travis W. Hill (M.S.I.E. & S.E.)
  - Judy Liaw (Masters)
  - Charles Liggett (Undergraduate)
  - Stephen W. Ormon (M.S.I.E.)
  - Derrick Pratt, (MBA)
  - Mona C. Shelton (Ph.D.)
  - Thiyagarajan Venugopalan (M.S.I.E)
  - Tai Chi Wu (Ph.D.)
- **Industry Reviewers**
  - Michael Bailey, General Electric Aircraft Engines
  - Michael Cronin, Cognition Corporation
  - John Fondon, Northrop Grumman (retired)
  - Peter Frederic, Tecolote Research, Inc.
  - Brian Glauser, Cognition Corporation
  - Joseph Jaworski, Pratt & Whitney
  - Mohsen Rezayat, SDRC
  - Steve Rogers, Acquisition Logistics Engineering
  - Don Shrader, TechniRep, Inc.
  - Ron Shroder, Frontier Technology, Inc.
  - Harmon Withee, Tecolote Research, Inc.

4. FOUNDATION DOCUMENT

Project Title: **Cost Evaluation/In situ Design Cost Trades**
Contract F33615-96-D-5608, Delivery Order No. 34

Contracting agency:
Anteon Corporation (for Air Force Research Laboratory)
5100 Springfield Pike, Suite 509
Dayton, OH 45431-1264
(937) 254-7950

Principal investigator: Allen G. Greenwood, Ph.D., P.E.
Department of Industrial Engineering
Mississippi State University (MSU)
Mississippi State, MS 39762
Phone: (662) 325-7216; Fax: (662) 325-7618
Email: greenwood@ie.msstate.edu

Project Team: (all from MSU)
Allen G. Greenwood, Ph.D., P.E.
Masoud Rais-Rohani, Ph.D., P.E.
Shunri Guo
Stephen Ormon
Chad Hymel

25 September 2000

4.1 Executive Summary

Affordability is a major product requirement and a driving force in today's defense and commercial business environments. Cost, especially life-cycle cost, is a primary component of affordability and a primary measure of design quality; and, as a result, assessing/evaluating cost is a key business process in most industries. In order to design affordable products, engineers need design decision-support tools that are an integral part of the design process and provide timely and relevant feedback on the cost of design alternatives. In general, this basic need is not being met. While tools exist and are being developed that address portions of the problem, it is often unclear how they are related to and work with other tools and technologies, especially as the design evolves over time. This project provides an initial approach to addressing and resolving these questions and issues. It provides an important first step toward effective cost evaluation during design.

Since this is the initial step towards the development of a comprehensive design decision-support tool, a significant portion of the project focused on defining user needs and system requirements. As a prerequisite to these activities, the product life cycle and systems development processes were defined and the general characteristics of cost-evaluation environments were identified. Similarly, preliminary definitions of the trade study and cost-evaluation processes, that an IDCT tool must support, were developed. These activities led to the formulation of objectives of an effective IDCT Tool and the identification of high-level functions of the Tool. All of these activities provided the foundation for identifying user needs and establishing system requirements. In addition, a preliminary concept of operation for the Tool was formulated.

A conceptual design for an IDCT was developed using an object-based approach. The design scheme focuses on the intelligent integration of cost elements (models, data, tools, etc.) in order to provide an effective decision support system for cost evaluation during design, especially during the conceptual and preliminary design phases. Based on the conceptual design for the IDCT Tool, commercial off-the-shelf (COTS) software was identified that could support the development of the Tool. In order to demonstrate feasibility of the approach and design, we developed a prototype of an IDCT Tool.

The project involved extensive literature and Internet searches for information on the design and cost evaluation processes and the types of models and technologies that would a part of an IDCT Tool. Information was also gathered at professional conferences and at meetings with personnel from industry that either are involved in the design and/or cost-evaluation processes or are involved with tools/technologies that support those processes. A portion of the conceptual design was tested and demonstrated through the development of a prototype.

All of the activities of the project provide a solid foundation for further development of the IDCT Tool. To that end, this project also developed a master plan for further development of the Tool in a subsequent project and identified potential team members that would be a part of the development.

4.2 Introduction and Background

Design is “a process of converting information that characterizes the needs and requirements for a product into knowledge about a product,” (Mistree *et al*., 1990) while satisfying a set of performance objectives and constraints. Design is driven primarily by a set of requirements that are typically based on some preconceived notion of quality.

Quality can be defined as “the extent to which a product responds to the demands of the customer and the marketplace” (Hisakazu *et al*., 1988). In this spirit and that of Garvin’s eight dimensions of quality (Garvin, 1984)], we consider performance, affordability, cost, manufacturability, reliability, etc. all to be elements of quality. For example, affordability – the explicit consideration of both a product’s performance and cost in order to develop a balance among requirements, costs, and budgets – is a major product requirement and a driving force in today’s defense and commercial business environments. In the case of defense systems, procurement requirements have shifted from performance at all cost to one ensuring superiority, yet being affordable. In 1995, Paul G. Kaminski, Under Secretary of Defense (Acquisition & Technology) described DoD’s new more balanced “cost of performance” view where cost is a critical component of DoD system optimization and life-cycle cost is not simply an outcome as has been the case in the past but an *independent* variable in meeting the user’s needs. Similarly, in 1997 NASA Administrator Daniel S. Goldin announced a major initiative to significantly reduce the cost of future air travel by 25 percent in ten years and 50 percent in twenty years. The program focuses on, among other things, reducing manufacturing cost of aircraft through novel airframe structural design and manufacturing processes, as well as improving design tools to reduce life-cycle costs.

Just as it has been said that quality must be designed into a product, affordability as well must be *designed into* products, rather than “checked” after the fact. Designers and integrated product teams (IPTs) must explicitly address cost as an integral part of the design process, especially in early phases of design. While affordability must be addressed throughout a product’s life cycle, a large portion of a product’s affordability is obtained and maintained by identifying and assessing cost drivers and explicitly addressing process design and manufacturing operations early in the design process. A common rule of thumb in industry is that nearly 75 percent of a product’s cost is determined by the end of conceptual design; therefore, there is a brief “window of opportunity” to significantly effect affordability.

A major aspect of design involves an iterative process of deciding the values for variables that define a design and evaluating the resulting performance in terms of how well the design meets its intended needs and objectives. This trade-study process is the foundation for finding satisfactory or “optimal” designs. As indicated above, a key performance measure for evaluating the quality of a design, and the focus of this report, is the anticipated cost of a product.

Cost, especially life-cycle cost, is a primary component of affordability and a primary measure of design quality; and, as a result, assessing/evaluating cost is a key business process in most industries. In order to provide adequate evaluation of candidate designs, cost measures that are used in design trade studies and design decision-making processes must explicitly consider the product’s form, material, manufacturing processes, use and operation, support concepts, etc. In order to *design in*, rather than “check” affordability after the fact, designers and integrated product teams (IPTs) must explicitly address cost as an integral part of the design process.

While many factors that are decided upon in the design process – e.g. size, materials, dimensional tolerance, part count etc. – are directly related to manufacturing and cost performance, such factors are often not addressed during the design process in a manner that enhances the quality of the final product. As Yoshimura (1993) describes, "in usual design optimization, consideration is only given to the improvement of the product performance … the product manufacturing and cost factors concerning manufacturing processes are scarcely considered." If cost performance measures are used at all in design trade studies, they mostly address a product's form and material, with little to no regard for the process that will be used to produce the product. Also, many existing cost models are based on variables that often are neither related to product features and design variables, nor characteristics of manufacturing processes; i.e., they are not sensitive to the factors that directly impact product cost. This is illustrated by an example from aircraft design (Fondon *et al*., 1996) where weight is commonly used as an early predictor of cost. As illustrated in Figure 4-1, since the skins in this horizontal torque box example comprise 66 percent of the item's weight, one might concentrate a cost improvement effort on the skins. However, the ribs, while comprising only 10 percent of the item's weight, contribute to 36 percent of the item's manufacturing cost and approximately half of the item's parts, tools, and labor cost.

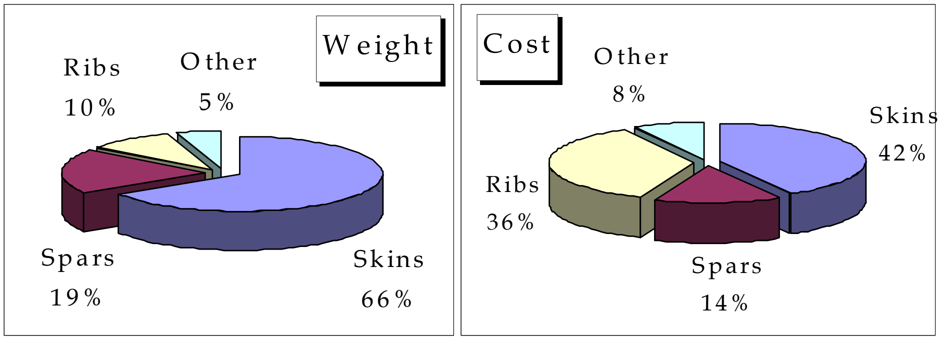


Figure 4-1: Weight-driven cost models do not support meaningful design trades.

In order to significantly impact design, cost, schedule, and risk must be assessed during design in a manner similar to the way aerodynamics, weight, etc. are assessed. In essence, designers need a "balanced scorecard" to guide the design process. Therefore, advanced design technologies must encourage and support the transformation of customer requirements into a product by providing cost/manufacturability measures of the design and concurrently considering the production process in conjunction with the product's intended form and material.

Unfortunately, such cost/manufacturability evaluation technologies are not readily available today. Historically the treatment of cost, both manufacturing and life-cycle, early in the design has been insufficient. As detailed in among others, (Dean, 1998), (Greenwood, 1996), and (Thomas, 1994), this is primarily due to: (1) an overarching focus on performance at all cost, (2) the lack of adequate design guidelines, such as supportive product and process evaluation data, models, processes, and tools that link design variables to cost measures, and (3) the lack of means to integrate these considerations into the product design process. Significant advances have been made in the past few years to develop technologies that both directly and indirectly support affordability and cost evaluation. Since cost-evaluation tools and technologies for supporting design have just begun to evolve, much needs to be done in order to integrate cost evaluation into the design process and elevate cost evaluation to a comparable level of application as other types of engineering analyses.

In order to design affordable products, engineers need design decision-support tools that are an integral part of the design process and provide timely and relevant feedback on the cost of design alternatives. In general, this basic need is not being met. While tools exist and are being developed that address portions of the problem, it is often unclear how they are related to and work with other tools and technologies, especially as the design evolves over time. A fundamental problem is that there is no unifying framework that describes the design/cost-evaluation processes over the product life cycle. This is closely related to the problems associated with the lack of a general understanding of the: (1) design decision-making processes that utilize cost, (2) cost-evaluation processes, especially those in support of design decisions, and (3) relationships between design processes and cost-evaluation processes over the product life cycle. Another related problem is that cost-evaluation methodologies often do not provide relevant and timely support to designers and integrated product teams (IPTs), especially early in design, because the inputs they require are not closely linked to design variables and the methodologies do not explicitly address production and operations/support considerations.

While it is all well and good to say that cost evaluation must be an integral part of design, it is not clear what this really means. For example, what is required to explicitly address cost and affordability issues early in the design process? As with any development effort, requirements are generated based on an understanding of the characteristics of the general cost evaluation operating environment. Similarly, once the requirements are defined, solutions for meeting the requirements are generated and implementation issues are addressed. Just as one would not design a new product in isolation, without understanding customer needs, it would be premature to attempt to integrate cost evaluation technologies into the design process without a fundamental understanding of the problems, issues, and needs and clearly defining the process involved. This report provides an initial approach to addressing and resolving these questions and issues. It provides an important first step toward effective cost evaluation during design.

4.3 Objectives and Approach

This project is a first step towards designing, developing, and deploying an In situ Design Cost Trades Tool. Such a tool would be an open, comprehensive system for supporting the design decision-making processes. It would provide an effective means for addressing the cost performance of candidate designs throughout the product life cycle. The system would be intended to: (1) allow the designer and/or IPT to adequately evaluate cost as an integral part of the design process, i.e., *as* the design evolves; (2) capture evolving design experiences, not only by retaining design cost information, but to reuse, exploit, leverage, and ultimately learn from the experiences; and, (3) provide a means to study and analyze, and ultimately better understand, the impacts of design and programmatic changes on product/process costs, i.e., identify and understand *what* and *how* these variables influence cost.

As a first step, the intentions of this initial phase of developing an effective IDCT Tool, as defined in the Statement of Work, was to:

a) identify user needs and requirements for an IDCT system,
b) perform a survey of commercial-off-the-shelf (COTS) software to use to meet the functional requirements of an IDCT system,
c) develop a master plan and schedule for the next phase of work on the IDCT system,
d) develop and demonstrate a prototype that establishes the feasibility of the approach,
e) identify a cross-functional technical review board (TRB), and

f) demonstrate the IDCT prototype to, and obtain feedback from, customers, stakeholders, and potential beta sites for Phase 2.

From a systems engineering perspective, this project is positioned in the conceptual design phase but includes all systems engineering steps within this phase, e.g. requirements analysis, functional analysis, synthesis, etc. A major focus of the project is to provide early front-end analyses and initial system requirements definition. A key aspect of the project was to identify the necessary components that need to be included within an IDCT system and understand how the components fit together. This requires a top-down approach and view cost evaluation in its broadest terms.

Cost evaluation, especially as an integral part of conceptual and preliminary design, takes places in an uncertain, dynamic environment. As a means to deal with the inherent complexity and to provide effective decision support, the IDCT system needs to be developed and managed from a systems perspective; i.e., it should to be *engineered* or systematically defined, designed, developed, and implemented. This requires a clear framework and set of processes that will, among other things, drive data and model requirements, and not visa versa (as has too often been the case). The systems approach to developing the IDCT system provides the opportunity to define and model cost evaluation and cost management processes so that they reflect the way cost evaluation *should* be conducted in order to support design. The motivations for defining and modeling cost-evaluation processes (based on Vernadats' motivation for enterprise modeling (Vernadat, 1996) are to: provide better understanding and uniform representation, provide a basis for analysis, support design or redesign of process elements, manage complexity, capitalize on acquired knowledge and facilitate its re-use, rationalize and secure information flows, enhance communication among functional entities, and build a common culture and shared vision.

The systems approach that is used in this project is illustrated in Figure 4-2. Prior to developing the requirements for an IDCT system, even at the conceptual level, it is critical to have a clear definition of the environment in which the system must operate. As shown at the top of Figure 4-2, there are three activities that define the environment and provide the bases for the system requirements. The three activities are defining the objective of a life-cycle cost evaluation environment, identifying the characteristics of the operating environment, and defining the design and cost-evaluation processes that the IDCT system will support. While the system will focus on the conceptual and preliminary design phases of the product life cycle, it is expected to operate over the entire life cycle. Similarly, it will obtain and use information from all phases of the life cycle, even during the conceptual and preliminary design phases.

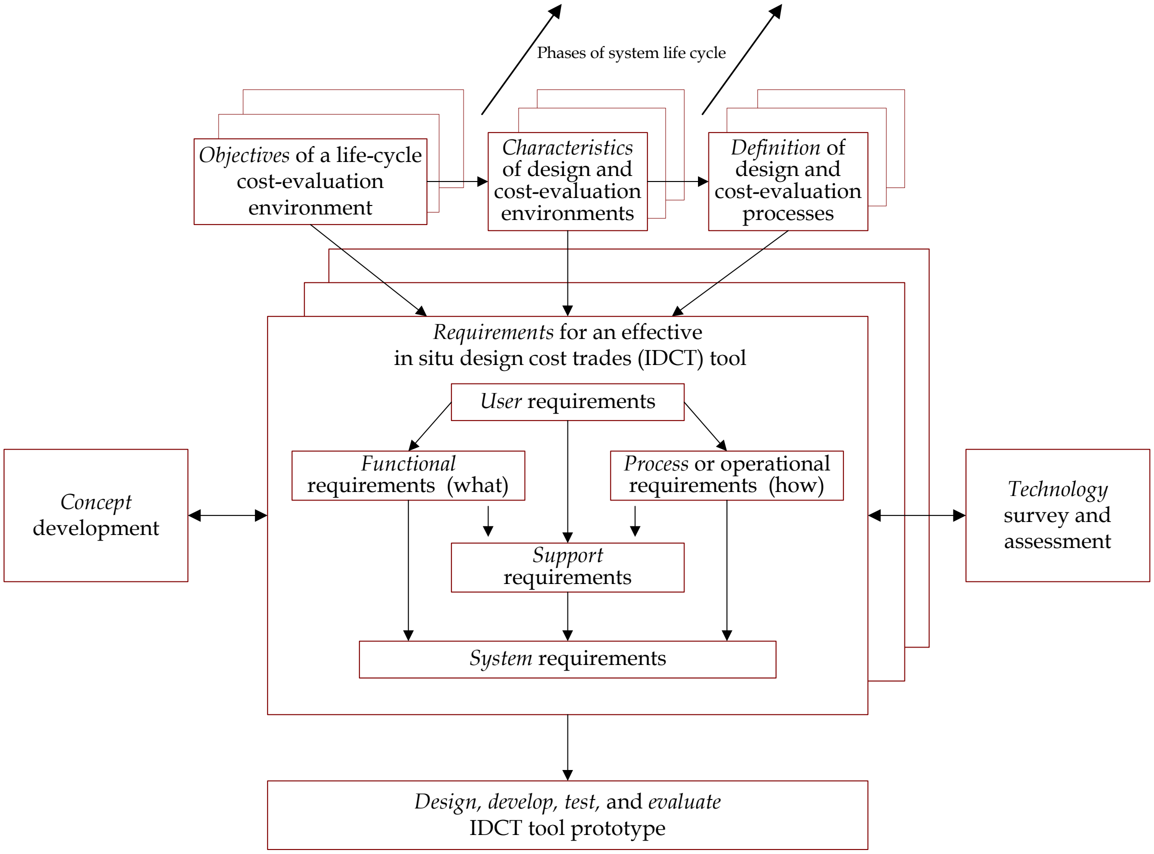

Phases of system life cycle
Objectives of a life-cycle cost-evaluation environment
Characteristics of design and cost-evaluation environments
Definition of design and cost-evaluation processes
Requirements for an effective in situ design cost trades (IDCT) tool
User requirements
Functional requirements (what)
Process or operational requirements (how)
Support requirements
System requirements
Concept development
Technology survey and assessment
Design, develop, test, and evaluate IDCT tool prototype

Figure 4-2: Systems engineering approach to conceptual design of IDCT Tool.

It is only once the objectives have been defined, the operating environment understood, and the processes defined, that user needs and requirements can be formulated. As shown in Figure 4-2, user requirements are composed of two primary components, functional requirements (what is to be done by the IDCT system) and process or operational requirements (how the functions must be accomplished). In addition, support requirements are identified; i.e., those that facilitate, enable, and sustain the primary activities that provide cost evaluation support during design.

Requirements are also formulated as the concept for the system is developed and concurrently with the technology survey and assessment. Defining requirements is a highly iterative process, changing frequently as the literature and user surveys provide better problem definition, concept development matures, and technology surveys identify enabling capabilities. Requirements will also evolve as the prototype is developed and demonstrated, and will continue to change as the as the actual system is developed. As system definition and development evolve, the developer better understands user needs; similarly, and users better understand the capabilities that are available to enhance and support their work.

The requirements for an IDCT system are a primary output of this contract; they are the initial set of requirements for a comprehensive cost-evaluation decision-support tool.

4.4 Description of Effort

As stated in the previous section, this project is a first step toward developing an effective IDCT Tool and involved six elements that were specified in the Statement of Work (SOW). Those work elements are listed below, along with the key activities that were performed within each element. Each key activity is identified in the list below by a letter in parentheses under its corresponding SOW element; each activity is discussed in detail in Section IV: Results. As indicated by the list of activities within each SOW element, most of the effort on this project was focused on the first and fourth elements – identifying requirements and developing a prototype, respectively.

1) Identify user needs and requirements for an IDCT system,
   a) Definition of the product life cycle and systems development processes, from the literature.
   b) Definition of the trade study process.
   c) Definition of cost evaluation processes.
   d) Objectives of an IDCT Tool.
   e) Functions of an IDCT Tool.
   f) Requirements for an IDCT Tool.
2) Perform a survey of commercial-off-the-shelf (COTS) software to use to meet the functional requirements of an IDCT system,
   a) Identification of potential COTS software tools/technologies that meet development needs of IDCT system from the literature, but mostly from the Internet.
   b) Review of reliability prediction during conceptual design. Since the IDCT system is to address life-cycle cost and since reliability is a major driver in life-cycle cost, a thorough review of the reliability literature was conducted to identify means for assessing reliability during conceptual design.
3) Develop a master plan and schedule for the next phase of work on the IDCT system.
   a) Master plan for subsequent development activities
4) Develop and demonstrate a prototype that establishes the feasibility of the approach,
   a) Definition of the conceptual foundation for an IDCT Tool
   b) Definition of an intelligent integration scheme
   c) Definition of a concept of operations for an IDCT Tool
   d) Definition of a prototype of an IDCT Tool
5) Identify a cross-functional technical review board (TRB), and
   a) Definition of potential members of a TRB and their roles.
6) Demonstrate the IDCT prototype to, and obtain feedback from, customers, stakeholders, and potential beta sites for Phase 2.

Due to an issue discussed in the prototype portion of Section IV, the prototype is not fully capable and as a result has not been demonstrated. We will provide the prototype, at no additional cost, as soon as the problems have been solved, most likely by November 1, 2000.

Figure 4-3 provides flow type view of the project's primary activities. While the project appears to be linear, it is only a general representation; there was considerable overlap in the activities and considerable looping back to prior activities. Preliminary definitions of the design and cost evaluation processes were essential first steps. The processes provided concepts of operation (ConOps) or illustrations of how the IDCT Tool would be used and how it needed to interface with the design and cost evaluation processes.

The ConOps provided the foundation for the conceptual design of the IDCT Tool. Part of the design was defined in more detail and tested through the development of a software prototype. The ConOps, design, and prototype all provided insight into the type of individuals that could best serve subsequent development partnering and through TRB membership.

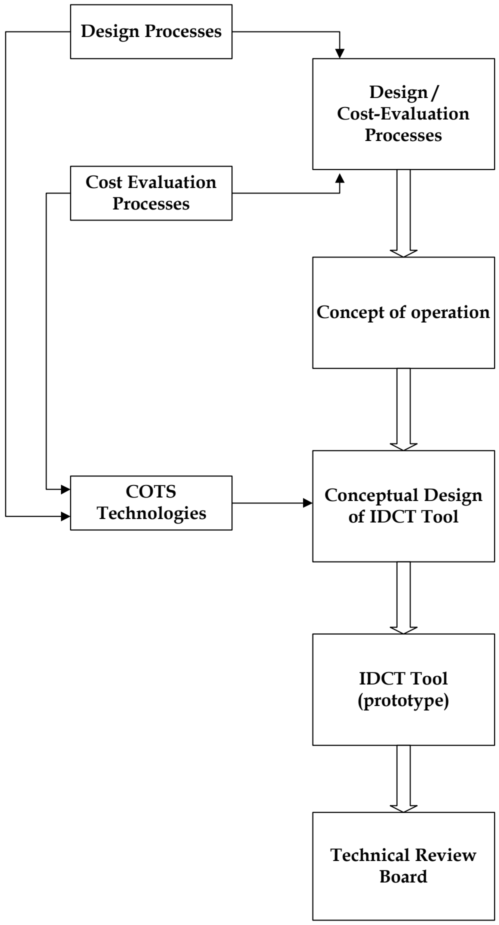


Figure 4-3: Primary project activities and project flow.

The activities discussed above address specific SOW elements. In addition to these specific activities, the project involved more general activities, such as library literature searches, Internet searches, attendance at professional conferences, and meetings with industry personnel in order to discuss design and cost processes and tools and technologies that would be utilized by an IDCT system. A partial list of the results of the literature search is provided in Appendix 2, Bibliography.

Table 4-1 summarizes the conferences that were attended as part of this project. The primary purpose(s) for attending each conference is noted in the last column in terms of the SOW element number, i.e., which primary activities the conference supported.

Table 4-1: Conferences attended as part of project

Title	Sponsor(s)	Location / Date	SOW
Open Workshop for Decision-Based Design	NSF (National Science Foundation) and ASME (American Society of Mechanical Engineers)	Las Vegas, NV Sept. 12, 1999	1
Affordability and Cost Modeling Workshop	Air Force Research Laboratory, Materials and Manufacturing Directorate	Fairborn, OH Nov. 1 - 2, 1999	1, 2, 4, 5, 6
Joint Cost Management Societies Symposium	AACE, ASPE, ISPA, PCEA, SAVE, SCEA (see abbreviation definitions below)	Chicago, IL Nov. 8 – 10, 1999	1, 2
Reliability and Maintainability Symposium	IEEE Reliability Society, Society of Reliaability Engineers, Society of Logistic Engineers, etc.	Los Angeles Jan. 22 - 25, 2000	1, 2
Theoretical Foundations for Product Design and Manufacture	Gordon Research Conferences	Plymouth, NH June 11 - 16, 2000	1, 2

AACE (Association for the Advancement of Cost Engineering), ASPE (American Society of Professional Estimators), ISPA (International Society of Parametric Analysts), PCEA (Professional Construction Estimators Association), SAVE (Society for the Advancement of Value Engineering), SCEA (Society for Cost Estimating and Analysis)

Table 4-2 summarizes the meeting with industry personnel that were conducted as part of this project. The primary purpose(s) for the meeting is noted in the last column in terms of the SOW element number, i.e., which primary activities each meeting supported.

Table 4-2: Industry meetings attended as part of the project.

	Location / Date	SOW
Acquisition Logistics Engineering Charles Coogan	Columbus, OH Feb. 1, 2000	1, 2, 5, 6
DFV Group, Inc. Edwin Dean	Minneapolis, MN Sept. 10,1999	1, 5, 6
Frontier Technologies Ron Shroder	Beavercreek, OH Jan. 31, 2000	1, 2, 5, 6
General Electric Aircraft Engines Gene Wiggs and Dr. Michael Bailey	Evendale, OH June 26, 2000	1, 2, 5, 6
James Gregory Associates, Inc. Dr. James R. Brink	Columbus, OH Feb. 1, 2000	1, 2, 5, 6
Knowledge Base Engineering Sam Nusinow	Centerville, OH Feb. 2, 2000	1, 2, 5, 6
Brian Noton (formerly of Batelle)	Columbus, OH Feb. 1, 2000	1, 5, 6
StraTech, Inc. Dr. Richard Thomas	Centerville, OH Nov. 3, 1999; Feb. 2, 2000; June 27, 2000	1, 5, 6
Structural Dynamics Research Corporation Dr. Mohsen Rezayat	Milford, OH Feb. 2, 2000; June 26, 2000	1, 2, 5, 6
Tecolote Research, Inc. Harmon Withee	Beavercreek, OH Jan. 31, 2000	1, 2, 5, 6

4.5 Results

Each of the key activities that are associated with a SOW element, as identified in the previous section, are defined and discussed in the following subsections.

4.5.1 Definition of the Product Life Cycle and Systems Development Processes

The IDCT system should function in all phases of the product life cycle; however, the focus will be on the conceptual and preliminary design phases. As described earlier, it is within these phases that the most benefit should result from effectively evaluating the cost of alternative designs as an integral part of the design trade study processes. In addition, the IDCT system may need to access information from any phase of the life cycle, regardless of the phase in which it is evoked. Also, the life cycle and systems development processes are the foundation on which the design processes and its associated activities are based, as well as the cost evaluation processes and activities that support design decision making. Therefore, for these reasons, it is necessary to adequately define the life cycle and systems development processes. We were encouraged at the first project review to conduct this investigation.

Based on a review of the design literature, six methodologies were selected, reviewed, and summarized. Table 4-3 provides a brief summary of the activities associated with each methodology; a more detailed description of the activities is provided in Appendix 1. By considering six methodologies, we believe we have obtained a complete view. The product design methodologies were compared and composite methodology was derived. It is shown in abbreviated form in the last column in Table 4-3 and in detail in Appendix 1.

Table 4-3: Product Design Methodology Comparison

Ulrich& Eppinger, 1995	Magrab, 1997	Pugh, 1991	Rouse, 1991	Dant & Ken-singer, 1997	Blanchard & Fabrycky 1998	Composite
	1. Establish Company Strategy		Recognition			1. Establish company strategy
	a. Establish cross-functional team					a) Establish cross-functional team
1. Concept Development	2. Establish customer needs	1. Examine market		1. Innovation Stage	1. Conceptual Design	2. Identify customer needs
a) Identify needs of customer						a) Establish target specification
	3. Define Product					3. Define product
	a) Determine feasibility of product and compare to strategy					a) Determine feasibility of product and compare to strategy
	b) Competition identified and benchmarked					b) Identify and benchmark competition
b) Establish target specifications	c) Adjust product def. as required and draw up preliminary product specs.	2. Specifications				c) Establish target specifications
		3. Concept Design	Formulation	2. Development Stage	a) Def. Of Need	3. Develop concept
c) Analysis of competitive products						a) Generate feasible designs
d) Generate and evaluate alternative product concepts	4. Generate Feasible Designs	a) Generate solutions to meet stated need		a) Incubation phase		b) Evaluate feasible designs
	5. Evaluate Feasible Designs	b) Evaluate solutions				c) Explore the most promising concepts and try different configurations
	a) Explore the most promising concepts and try different configurations					d) Select single concept for further development
e) Select single concept for further development	b) Choose best configuration					f) Refine specification
f) Refine specification						4. Market and research phase
g) Economic analysis			Analysis	b) Market and research phase	b) Feasibility study	a) Feasibility/ economic study
h) Project planning					c) Advance product plan	b) Project planning
2. System Level Design	c) Generate drawings and prototypes		Formulation		3. Preliminary Design	5. System Level Design

Ulrich& Eppinger, 1995	Magrab, 1997	Pugh, 1991	Rouse, 1991	Dant & Ken-singer, 1997	Blanchard & Fabrycky 1998	Composite
a) Def. of product architecture						a) Define product architecture
b) Division of product into subsystem						b) Divide product into subsystems
c) Define final assembly scheme						c) Define final assembly scheme
			Analysis		a) Functional analysis	d) Perform system functional analysis
					b) Preliminary synthesis and alleviation of design criteria	e) Preliminary synthesis and alleviation of design criteria
					c) System optimization	c) System optimization
					d) System synthesis and definition	d) System synthesis and definition
d) Outputs: “layout” of product						e) Output: “layout” of product
3. Detail Design		4. Detail design	Formulation	c) Engineering or “fit” phase	4. Detail Design and Development	6. Detail Design
a) Complete specifications					a) System-product design	a) Complete specifications
b) Establish process plan and tooling						b) Establish process plan and tooling
c) Output: control documentation				d) Documentation phase		c) Output: control documentation
4. Testing and refinement a) Alpha prototype b) Beta prototype	d) Evaluate prototypes compared to preliminary product specifications		Synthesis		b) System prototype development	d) Develop system prototype
					c) System prototype test and evaluation	e) System prototype test and evaluation
	6. Process Design		Formulation			7. Process Design
	a) Design manufacturing processes and assembly procedures					a) Design manufacturing processes and assembly procedures
				e) Procurement phase		8. Procurement phase
				f) Distribution phase		9. Distribution Phase
5. Production Ramp-Up				g) Product launch phase		10. Production Ramp-Up
a) Train work force						a) Train work force
b) Transition into production	7. Manufacture and Assemble	5. Manufacture	Fabrication	3. Production Stage	5. Production and/or Construction	b) Transition into production

Ulrich& Eppinger, 1995	**Magrab, 1997**	**Pugh, 1991**	**Rouse, 1991**	**Dant & Kensinger, 1997**	**Blanchard & Fabrycky 1998**	**Composite**
					a) System evaluation	11. Production
					b) Modication for C.A. and/or prod. Improvement	a) System evaluation
					6. Utilization and Support	b) Modification for C.A. and/or prod. improvement
					a) System evaluation, analysis, and evaluation	12. Utilization and Support
					b. Modification for C.A. and/or for product development	a) System evaluation, analysis, and evaluation
	8. Market Product	6. Sell				b) Modification for C.A. and/or for product development
				4. Obsolescence Phase	7. Phaseout	13. Sell
						14. Phaseout

4.5.2 Characteristics of Cost-Evaluation Environments

A critical and essential prerequisite to developing an effective IDCT Tool, is the identification and specification of needs. However, before the needs can be articulated, it is necessary to characterize the environment in which this tool will be used. Therefore, this section defines the general characteristics of typical cost-evaluation environments. While any particular environment may not exhibit all of the characteristics listed below, and exceptions to the following statements do exist, the characteristics are quite common and widespread. Specific cost-evaluation environment characteristics are classified into general categories; the specific characteristics are denoted by a letter designation within a general category.

1) **Ambiguous --** Cost is a common term; however, its notion and the processes associated with developing cost estimates are not well understood, especially by designers. The issues associated with ambiguity relate to *what* is being assessed; i.e., there is often confusion on *what* is meant by the cost measure(s).
   a) Cost is an abstract and intangible entity; i.e., it cannot be seen, weighed, etc.
   b) There is a lack of consistency in definitions and use of terminology.

2) **Disparate/cross-functional --** Cost evaluation involves the understanding and use of information, data, models, expertise, etc. from across an organization. In many cases today, cost evaluation even crosses company boundaries, as is in the case of virtual organizations, strategic partnerships, etc. There are three primary *user groups* involved in cost evaluation: product designers, domain experts (cost and financial analysts, manufacturing engineers, materials specialists, etc.), and managers (design, product, and process).
   a) Each user group has different needs and roles in assessing the cost of a product. The focus of each group is either on producing/providing or using information for cost evaluation.
   b) There are numerous interactions among designers, domain experts, and managers, with regard to the development and use of cost estimates; the relationships are not well defined.
   c) Cost evaluation is a complex set of related activities that involve heterogeneous data, models, knowledge, and organizations.
   d) *Cost elements* are the data, models, knowledge, etc. that are needed to assess the cost of an item. They are widely distributed and are created and maintained (i.e., reside) in many physical locations, hosts, and organizations.

3) **Longitudinal--** Cost evaluation is performed in all phases of a product's life cycle. The need for and value of cost evaluation information varies over the product's life cycle; it is often of most value early in the product life cycle.
   a) Cost evaluation processes vary over the product life cycle.
   b) Processes change because cost evaluation *approaches* vary over the product life cycle. As described in (MTC, 1995) there are four basic cost-evaluation approaches: (1) judgment – use of expert opinion of one or more qualified experts in the area to be estimated, (2) parametric -- mathematical expressions, often referred to as cost estimating relationships or CERs, that relate cost to independent cost driving variables, (3) analogy – use of actual costs from a similar existing or past programs with adjustments for complexity, technical or physical differences in order to derive a new system estimate, and (4) grass roots, also referred to as "engineering build up" or "detailed estimate" are performed at the functional level of the Work Breakdown Structure.
   c) Approaches change because the availability of information used in cost evaluation varies over a product's life cycle. As noted above, the most valuable time for cost evaluation is early in the

design process; however, this is when the least information is available, and when the information is available, it is the least certain.
d) As a result of changing processes, approaches, and information availability, the type of cost evaluation tools and technologies vary over the product life cycle.
e) In addition to tracking cost performance for a specific project over time, knowledge from similar, previously analyzed projects should be utilized to the fullest extent possible. This does not just include the final designs, but modifications and alternatives that were considered and subsequently not adopted.

4) **Time-sensitive** -- Performance of the design, especially in terms of cost, needs to be assessed *as* the design evolves. Immediate feedback and guidance on alternative designs reduce design time and expands the design space.
   a) Cost evaluations are most effective and of most value when they are performed in near "real-time" in order to provide feedback on design alternatives as soon as possible. This is especially important in the early design phase
   b) Cost evaluation is often a time-consuming, cumbersome process, particularly in the early design phase. This is due to many factors, lack of process and structure, lack of automation of cost evaluation approaches and information acquisition, lack of readily available information, etc. In early design, it is often due to the application of models that require too much information and too much time to develop to be useful in early design. Often designs evolve faster than estimates. On the other hand, parametric models are generally not very sensitive to meaningful design variables. There is little continuity between estimates, e.g. used to bid, allocation, and possibly reallocation.

5) **Technologically non-homogeneous designs** – The design of complex systems are composed of components that are at different levels of technological maturity, from common off-the-shelf items to those that are just emerging from the laboratory. This is true even during the conceptual and preliminary design phases, i.e., not all components are vague, and many may actually be well understood. Since the type and amount of information that are available on the components vary widely, different models must be used to assess their cost.
   a) The cost of a design is an aggregation of costs obtained from multiple models that are based on varying degrees of design resolution and information fidelity.
   b) The cost of the design should be based on the most appropriate and most current information available, e.g. even in conceptual design, costs of off-the-shelf components should be based on actual production or field data.

6) **Dynamic** — Change is one of the few constants in most design environments.
   a) The product being designed often changes rapidly as a result of changing customer needs, alternative designs being generated, etc.
   b) Elements of cost evaluation (data, models, etc.) change over time due to a rapidly changing general business environment, emerging manufacturing technologies, new materials, etc.

7) **Uncertain** – The design environment, especially in early phases, contains many unknowns and uncertainties.
   a) All participants in a cost evaluation of a new product early in the design process operate in a highly uncertain environment. However, just as the primary user groups have different cost evaluation needs and roles in the evaluation processes, the nature of their uncertainty varies as well. Therefore, even though there may be overlap, uncertainty needs to be addressed from the three primary user groups' perspective (product designers, domain experts, managers).

8) Knowledge based -- The foundation of cost evaluation is not just data. It is extensively built upon insights, experiences, procedures, information, etc. that are drawn from many sources.

   a) Successful cost evaluation depends upon effectively assimilating and integrating diverse and disparate knowledge.
   b) It is often unclear in advance what knowledge is needed, when it is needed, in what format, and where it resides. Defining the cost evaluation processes that support the design trade study processes will identify these knowledge needs.

9) **Process based** – Processes are the way work gets done. As discussed above, cost evaluation involves a series of cross-functional processes. However, these processes are often not identified or defined and as a result in confusion and/or a lack of understanding. While heavily related to many of the characteristic described above, the lack of a process focus on cost evaluation is so significant that it demands its own category.
   a) Cost evaluation is a complex set of related activities that involve distributed data, models, knowledge, and organizations.
   b) The manner in which design information is translated into cost estimates varies widely, even in the same organization; it is rarely codified into written procedures. Basically, cost evaluation processes are not well defined and are not well documented; both the development and use of cost evaluations are not viewed as processes.
   c) Design decision-making processes are not well defined. Similarly, the needs of the decision makers are not defined; i.e., it is not clear what is needed to effectively support and enhance design decisions. As a result, it is not clear where, when, and how cost evaluation technologies can and should be integrated into design processes.

10) **Model centric** – Cost evaluation is heavily based upon the development and use of models. They are the primary means for translating design characteristics into cost measures.
   a) Models support a variety cost evaluation needs and functions. As a result, they are very disparate in the depth and breadth of information they require and provide.
   b) Cost models are typically disjointed; i.e., cost evaluation of a systems requires the assimilation, synthesis, and aggregation of cost models.
   c) Models used in cost evaluations are widely distributed among numerous functional units throughout the organization and are often proprietary.
   d) Model selection and application depends on the degree of design resolution and information fidelity.
   e) Individual model capabilities and requirements are often not systematically defined.
   f) Multiple models may be applied to a single design component.
   g) The following identify common shortcomings of cost models, especially for models used during conceptual and preliminary design.
      i) Cost model drivers are not related to product features and design variables.
      ii) Cost models are not sensitive to characteristics of the operations and support processes that will be used to operate and maintain the product.
      iii) Cost models are not sensitive to characteristics of manufacturing processes.
      iv) Cost models are static, not as current as the available data.
      v) Cost models are based on manufacturing methods and processes that are no longer used.
      vi) Cost models do not adequately address the relationships between direct product/process costs and indirect costs.

11) Data relevancy and currency – Data are a key element in cost evaluation. In order to effectively support the design process, the data that are available must be relevant and current. All too often, the data that are the most useful for assessing the cost of candidate designs either do not exist, exist in an unusable format, or are not directly accessible to the users or models that need them.
   a) Data used in cost evaluations are widely distributed among numerous functional units throughout the organization.

b) Data reside in many formats on a variety of platforms that are often not compatible and/or not accessible.

c) Data requirements have not been explicitly defined. Due to the lack of definition of cost evaluation processes, data needs are either not defined at all or are defined too late to be of much value. A process focus will result in definitions of how cost evaluations *should* be performed; this will naturally lead to the identification and definition of the data that are needed to support the processes. Once the data needs are identified and defined, processes can be established for collecting and maintaining the requisite data. In the past legacy data systems did not track information to the required level of detail that is necessary to support of the design process. Significant advances in technology have greatly reduced the problems of data collection and management. However, the remaining challenge is deciding what is needed, when, by whom, and in what form. These specifications result from identifying and defining cost evaluation processes.

12) **Islands of Technologies** – Cost elements and the technologies used to create, process, and manage those elements are often developed and used to address specific problems. As a result, they are not generalized, reused, or made available for reuse in subsequent projects.

a) Cost evaluation tools and technologies are not well integrated, due mainly to their development and use in disparate, non-homogeneous environments and their proprietary nature.

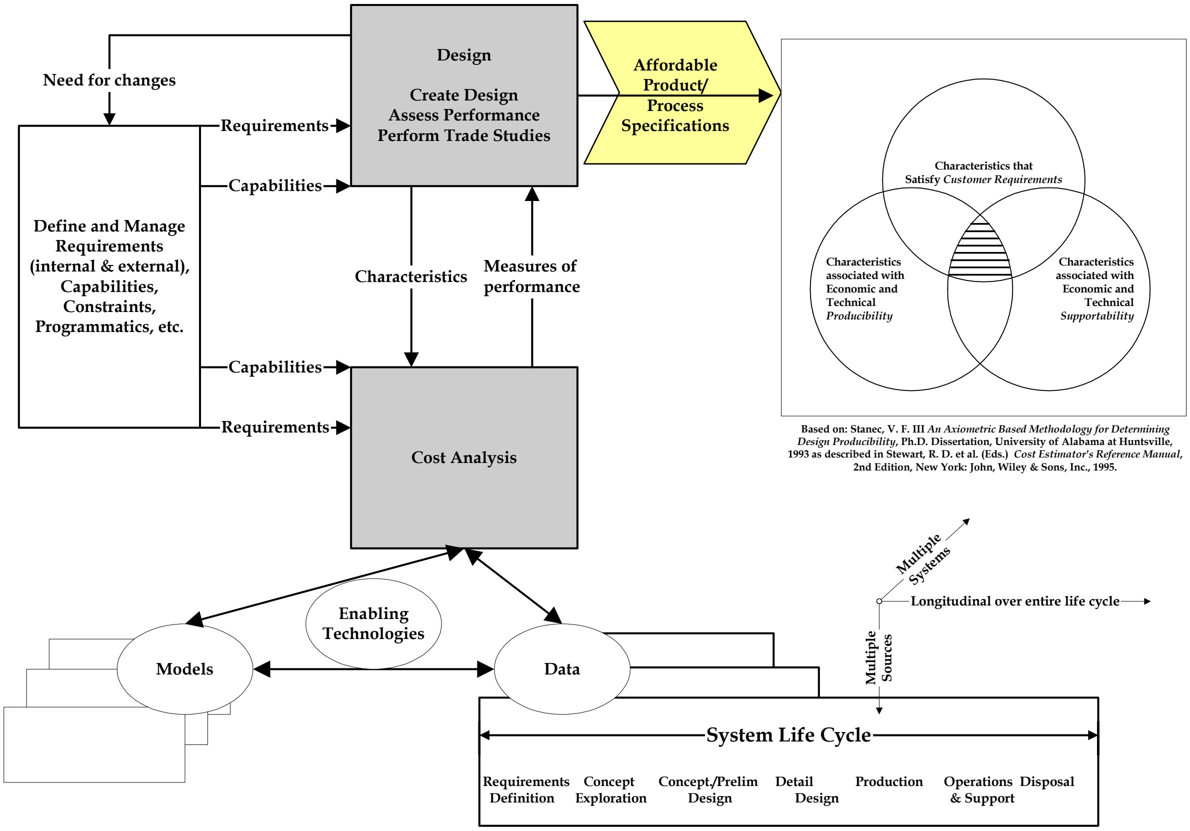
b) There is little sharing of common capabilities and services across comparable domains, applications, and technologies.

c) The use/re-use of cost-evaluation technologies is not well defined, e.g. it is often not clear how and when to use certain technologies. Identifying and defining structures for the cost elements would provide a means for classifying and describing the functions of specific technologies, defining their "world views" and roles in cost evaluation, and facilitate integration.

d) The development, use, and maintenance of cost-evaluation technologies is ad hoc, i.e., not based on an overall strategic plan. An overall plan would reduce the development of similar technologies, highlight missing capabilities and thus foster the development of technologies that would significantly advance cost evaluation support during design, and encourage the sharing of common cost-evaluation services and applications. The use of object-based technologies in conjunction with a master plan could effectively link many of the existing islands of technology.

4.5.3 Definition of the Trade-Study Process

Figure 4-4 provides a high-level, conceptual view of the design and cost evaluation processes and the relationships between them. Design is a creative process that transforms requirements and capabilities into affordable product/process specifications. The alternative solutions generated by the design process must be evaluated and their performance assessed. As shown in the three-circle diagram in the right-hand portion of Figure 4-4, the design process must generate feasible solutions that, as indicated by the shaded area at the intersection of the three circles, result in a set of characteristics that concurrently: (1) satisfy customer requirements, (2) are economically and technically producible, and (3) are economically and technically supportable. One of the primary measures of how well a design meets these three criteria is cost, especially life-cycle cost. Therefore, cost evaluation provides critical decision support to the design process, especially as a major aspect of trade studies.


Need for changes
Design
Create Design
Assess Performance
Perform Trade Studies
Affordable Product/ Process Specifications
Requirements
Capabilities
Define and Manage Requirements (internal & external), Capabilities, Constraints, Programmatics, etc.
Characteristics
Measures of performance
Capabilities
Requirements
Cost Analysis
Characteristics that Satisfy *Customer Requirements*
Characteristics associated with Economic and Technical *Producibility*
Characteristics associated with Economic and Technical *Supportability*
Multiple Systems
Longitudinal over entire life cycle
Multiple Sources
Enabling Technologies
Models
Data
System Life Cycle
Requirements Definition
Concept Exploration
Concept./Prelim Design
Detail Design
Production
Operations & Support
Disposal

Based on: Stanec, V. F. III *An Axiometric Based Methodology for Determining Design Producibility*, Ph.D. Dissertation, University of Alabama at Huntsville, 1993 as described in Stewart, R. D. et al. (Eds.) *Cost Estimator's Reference Manual*, 2nd Edition, New York: John, Wiley & Sons, Inc., 1995.

Figure 4-4: Design and cost evaluation are highly related transformational processes.

As shown in Figure 4-4, cost evaluation is also a transformational process, in that it transforms characteristics of a design, as well as requirements and capabilities, into measures of cost performance. As shown in the lower portion of Figure 4-4, the design characteristics are transformed to cost measures primarily through models and data. This occurs longitudinally across the system life cycle, involves multiple sources of data and models, and utilizes information from multiple comparable systems. This is made possible by the application and integration of numerous enabling technologies.

Figure 4-5 provides a high-level process-flow representation of the basic trade-study process. The primary activities of the process are indicated by the shaded boxes – develop/define candidate solutions, analyze candidate solutions, and evaluate candidate solutions. The numbers in the circles in each shaded box are reference number that are used later when each of these activities are broken into subactivities and are defined in more detail. The larger shaded box that nearly encompasses the primary activities represents the domain of a in situ design cost trades system. It would fully encompass the analyze component and partially involve the design/develop and evaluate activities.

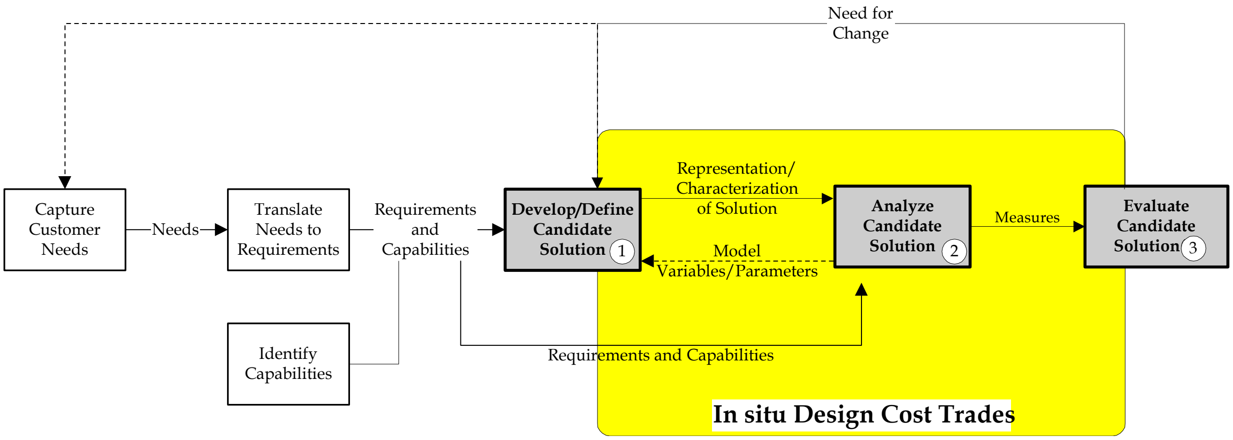

Need for Change
Capture Customer Needs
Needs
Translate Needs to Requirements
Requirements and Capabilities
Develop/Define Candidate Solution 1
Representation/ Characterization of Solution
Model Variables/Parameters
Analyze Candidate Solution 2
Measures
Evaluate Candidate Solution 3
Identify Capabilities
Requirements and Capabilities
In situ Design Cost Trades

Figure 4-5: A high-level representation of the general trade-study process.

As shown in Figure 4-5, representations and characterizations of design alternatives are transformed into measures that are used to evaluate the feasibility and performance of each candidate solution. Results of the evaluation drive the need for changes to a design in search of better alternatives. Both the develop/define and analyze activities utilize requirements and capabilities; however, are defined outside of the IDCT system.

4.5.4 Definition of Cost Evaluation Processes

Cost evaluation is composed of many processes. Most of the processes can be classified as either *using* or *producing* processes. "Using" processes focus on the use of cost-evaluation data, knowledge, models, etc.; whereas, "producing" processes focus on the provision and maintenance of cost-evaluation data, models, knowledge, etc. In order to develop an effective IDCT Tool, it is important to understand how the key stakeholders – designers, domain experts, and managers – interact with these two classes of processes. Relationships among the processes and the stakeholders are illustrated in Figure 4-6. The solid lines represent *primary* involvement; dashed lines represent *secondary* involvement. For example, designers and managers are primarily involved with use processes; i.e. they are primarily consumers of cost-evaluation data, models, knowledge, etc. Their secondary role is as suppliers of information to the producing processes, mostly in terms of expertise to help build models, enhance system effectiveness, etc. Domain experts have an opposite set of roles. Domain experts are the primary suppliers of cost-evaluation data, models, knowledge, etc.; therefore, their main involvement is with producing processes. Their secondary role is as consumer of information from primary users, mostly in the form of feedback on the effectiveness of the cost evaluation environment in enhancing trade studies and design decision making.

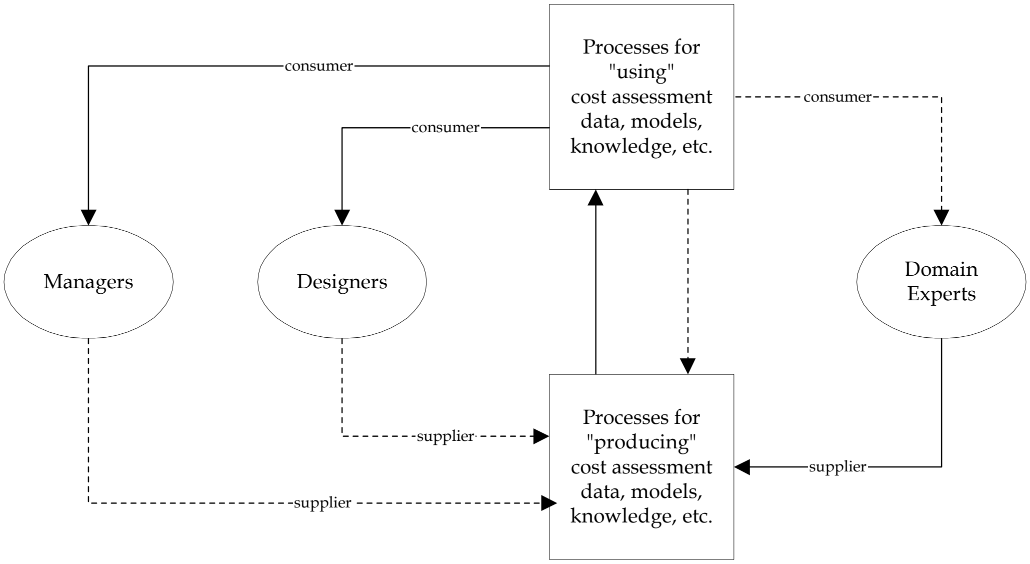

Processes for "using" cost assessment data, models, knowledge, etc.
consumer
consumer
consumer
Managers
Designers
Domain Experts
Processes for "producing" cost assessment data, models, knowledge, etc.
supplier
supplier
supplier

Figure 4-6: Cost evaluation involves “producing” and “consuming” processes.

Figure 4-7 provides a general representation of what is perceived to be a typical “using” cost-evaluation process for supporting design trade studies. It depicts the relationships between the design processes and the supporting cost models. In this representation, we show the product design and process (manufacturing) design processes as separate but linked processes.

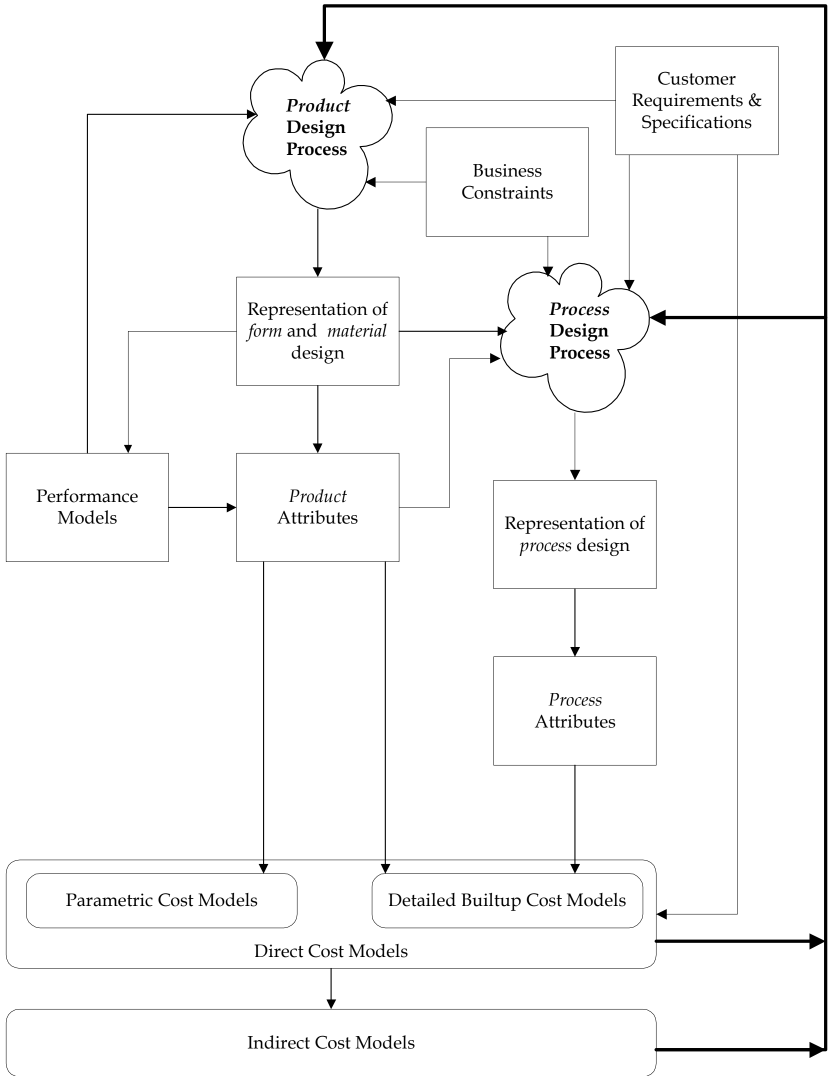


Figure 4-7: A generic "using" cost-evaluation process.

As shown in Figure 4-7, both design processes are driven by customer requirements and specifications and business constraints. In order to design affordable products, the process is also driven by performance measures and cost measures. Since the focus of this project is on cost evaluation, we separate cost from all other performance measures, even though cost itself is a measure, albeit a primary measure, of product/process performance.

The product design process, as shown in Figure 4-7, focuses on the form and material aspects of the product design. It results in representations of the design (e.g. sketches, CAD drawings, parts lists) that describe the product's material and form. These representations, along with models to assess the performance (e.g. weight, aerodynamics, failure), result in a set of product attributes. The attribute values are used in cost-evaluation models and in the process design process.

Process design, as shown in Figure 4-7, is based primarily on the product attributes, customer requirements, and business constraints. It results in representations of the process design (e.g. process plans, resource requirements) in terms of fabrication, assembly, and inspection; these subsequently result in a set of process attributes.

Cost models, as shown at the bottom of Figure 4-7, use customer requirements and product and process attribute values to assess the cost of the product. Results from the cost models provide feedback to the design process on the cost "performance" of the specified design. The two most common types of cost models are shown in Figure 4-7. Parametric models are high-level models based on few product attributes and often utilize difficult to quantify "complexity factors." On the other end of the spectrum, detailed cost models make heavy use of product and process attributes. Historically neither of these approaches has been effective for early design trade studies. The cost-evaluation process is only of value if it provides timely and meaningful information for design trade studies. As is well documented, most cost evaluation processes fall short of this need. While detailed cost models require too much information and too much time to develop to be useful in early design, parametric models are generally not very sensitive to meaningful design variables. Another shortcoming of both types of models is that historically they have dealt primarily with direct costs and have ignored indirect costs or have treated them as simple add-on factors.

The ultimate goal of this and subsequent research is to improve the current using cost-evaluation processes through the development of an IDCT Tool. The Tool will leverage existing models and technologies as well as facilitate and encourages better understanding by all stakeholders in the design process – designers, domain experts, and managers -- of the relationships between product/process attributes and cost.

4.5.5 Objectives of an IDCT Tool

The overall objective of an IDCT Tool is to provide an effective design decision-support capability that fosters, supports, and enhances design trade studies and decision-making processes throughout the product life cycle. The IDCT Tool should:

1. open the design trade space.
2. result in decisions that are based on knowledge drawn from as many sources as possible,
3. provide meaningful life-cycle cost evaluation, evaluation, and feedback, as an integral part of the design process, i.e., *as* the design evolves,
4. capture and utilize evolving design experiences as a means to learn from the experiences, and
5. foster a better understanding of the impacts of design and programmatic changes on product/process costs.

4.5.6 Functions of an IDCT Tool

The primary function of an IDCT Tool is to transform design and programmatic variables into cost measures, usually through the application of a series of models. An obvious function of the Tool is to provide estimates or *evaluations* of designs. In order to be consistent and expedient, the process must foster *learning* from past experiences. In addition, in order to effectively impact design, the IDCT Tool must provide means to seamlessly integrate evaluation and learning into the design process; i.e. it must include *realization* methodologies and technologies to encourage cost trades during design. The three fundamental functions that are associated with the transforming design variables into cost measures – evaluation, learning, and realization – are further defined below.

1. Cost evaluation -- An IDCT Tool allows the designer and/or IPT to quickly and adequately evaluate cost as an integral part of the design process, i.e., provide measures and meaningful feedback *as* the design evolves. As will be discussed later in this report, "traditional" cost estimating is considering a subset of evaluation.

2. Learning -- An effective IDCT Tool captures and provides means for utilizing evolving design experiences. This is not limited to only retaining design/cost information, but encourages reuse and exploits, leverages, and ultimately learns from each design experience and alternative that is considered. Ultimately, this function is aimed at better understanding the impacts of design and programmatic changes on product/process costs, i.e., identify and understand *what* and *how* these variables influence cost.

3. Realization -- In order to be an effective design tool, cost evaluation must be seamlessly integrated into the design process. It must be accepted and used by designers and IPTs. It most certainly must extensively utilize a variety of information technologies to not only implement analysis tools but foster consistency, improve communication, and enhance understanding.

The primary users of the IDCT Tool are product designers, domain experts (cost and financial analysts, manufacturing engineers, materials specialists, etc.), and managers (design, product, and process). By user, we mean anyone who provides information to or obtains information from the IDCT Tool. The IDCT Tool is expected to be used both in industry by practicing engineers and in academe by engineering students.

4.5.7 System Requirements for an IDCT Tool

System requirements must be based on user needs. However, before user needs can be identified, it is essential to understand the environment in which the users tool operate. Therefore, we begin with the characteristics of the cost-evaluation environment that were defined a previous section of this report. Each characteristic is mapped to a specific need(s), as shown in Table 4-4; subsequently, each need is mapped to system requirements, also shown in Table 4-4.

Table 4-4: Map of cost evaluation environment characteristics to IDCT Tool requirements.

ref no.	**Characteristics of cost evaluation environments**	**ref no**	**Needs to be met by, or through, the development of an IDCT Tool**	**ref no**	**Requirements for an effective IDCT Tool**
1	***Ambiguous; not understood***				
a	abstract and intangible				
b	inconsistent terminology and definitions	1	define and standardize terminology, structures, etc.	1	utilize dictionaries, common structures and hierarchies (e.g. cost, materials, processes, etc.)
2	***Disparate/Cross-functional***				
a	user groups have different needs and roles	2	define user group roles and needs in developing and using cost estimates	2	based on defined user group cost-evaluation roles and needs
b	interactions among user groups not defined	3 4 5	identify interfaces between tools / technologies and user types define processes for developing and using cost estimates identify means for collaborative work	3 4 5	provide effective, intuitive interfaces for user groups based on defined processes for developing and using estimates function in a collaborative environment
c	heterogeneous data, models, knowledge, etc.	6	provide means for disparate cost elements to communicate	6	link disparate cost elements
d	distributed data, models, knowledge, etc.	7	provide means for distributed cost elements to communicate	7	link distributed cost elements
3	***Longitudinal***				
a	processes for developing and using estimates vary over life cycle	4	define processes for developing and using cost estimates	4	based on defined processes for developing and using estimates
b	approaches (analogy, parametric, etc.) to developing cost estimates vary over life cycle	8 9	define information requirements to utilize approach define selection criteria for utilizing approach	8 9	support varied cost evaluation approaches provide guidance on approach selection
c	availability of information for developing cost estimates varies over life cycle	10 11	identify what information is available at various phases of life cycle identify where information resides and how to access it	10	provide access to all required information

ref no.	<u>**Characteristics**</u> **of cost evaluation environments**	**ref no**	<u>**Needs**</u> **to be met by, or through, the development of an IDCT Tool**	**ref no**	<u>**Requirements**</u> **for an effective IDCT Tool**
d	tools/technologies used to develop cost estimates vary over life cycle	12 13	define cost-evaluation tool / technology capabilities and requirements for use define selection criteria for cost-evaluation tool / technology	11 12	provide access to and support the use of various cost evaluation tools / technologies provide guidance on tool / technology selection
g	cost evaluation based heavily on historical knowledge	14	define the role of and means to use historical information	10	provide access to all required information
4	***Time sensitive***				
a	most value added when evaluation performed in near real-time	A L L	all needs support more timely cost evaluations	A L L	all requirements support more timely cost evaluations
b	time consuming, cumbersome process, especially early in design	A L L	all needs support better processes in order to provide timely cost evaluations	A L L	all requirements support better processes in order to provide timely cost evaluations
5	***Non-homogeneous designs***				
a	cost aggregation from multiple models and/or tools/ technologies based on varying degrees of design resolution and information fidelity	15 16	identify relationships among models, tools/technologies define means for utilizing multiple estimates	13 14	support application of multiple models to design components aggregate results from multiple models and design components
b	based on most appropriate and most current information	14	define the role of and means to use historical information	10	provide access to all required information
6	***Dynamic***				
a	rapid product changes	17 5	define roles of product information systems, e.g. PDM, CAD, requirements management identify means for collaborative work	15 5	utilize product information systems, e.g. PDM, CAD, requirements management function in a collaborative environment

ref no.	**Characteristics** of cost evaluation environments	ref no	**Needs** to be met by, or through, the development of an IDCT Tool	ref no	**Requirements** for an effective IDCT Tool
b	elements of cost evaluation (data, models, etc.) change over time	18 14 6 7	define means to manage changing cost elements define the role of and means to use historical information provide means for disparate cost elements to communicate provide means for distributed cost elements to communicate	16 10 6 7	manage cost element change provide access to all required information link disparate cost elements link distributed cost elements
7	***Uncertain***				
a	nature and degree of uncertainty varies by user group	19 2	identify how user groups handle uncertainty define user group roles and needs in developing and using cost estimates	17 18 2	provide means to explicitly address uncertainty in estimates provide means to facilitate sensitivity analyses and synthesis of multiple estimates based on defined user group cost-evaluation roles and needs
8	***Knowledge based***				
a	assimilation and integration of diverse and disparate knowledge from a variety of domains, e.g. materials, manufacturing processes	20 14 4	identify sources and uses of knowledge define the role of and means to use historical information define processes for developing and using cost estimates	19 10 4	provide access to knowledge sources and means to process and use knowledge provide access to all required information based on defined processes for developing and using estimates
b	Unclear in advance what knowledge is needed, when, in what format, and where it resides	4 2	define processes for developing and using cost estimates define user group roles and needs in developing and using cost estimates	4 2	based on defined processes for developing and using estimates based on defined user group cost-evaluation roles and needs
9	***Process based***				
a	complex activities involving distributed data, models, knowledge	A L L	all needs are based on complex cost evaluation processes	A L L	all requirements are based on complex cost evaluation processes

ref no.	**Characteristics** of cost evaluation environments	ref no	**Needs** to be met by, or through, the development of an IDCT Tool	ref no	**Requirements** for an effective IDCT Tool
b	design decision-making processes not well defined; similarly, the needs of the decision makers are not defined.	21	identify and define design decision-making processes relevant to developing and using cost estimates	20	based on defined design decision-making processes relevant to cost evaluation
c	translation of design information to cost estimate varies widely; cost evaluation processes not well defined	2 4	define user group roles and needs in developing and using cost estimates define processes for developing and using cost estimates	2 4	based on defined user group cost-evaluation roles and needs based on defined processes for developing and using estimates
10	***Model centric***				
a	models support variety of cost evaluation needs; disparate	6	provide means for disparate cost elements to communicate	6	link disparate cost elements
b	cost models disjointed	22 16	define means to assimilate, synthesize, and aggregate cost models define means to manage changing cost elements	21	provide means to assimilate, synthesize, and aggregate cost models
c	models used in cost evaluation widely distributed, proprietary	7	provide means for distributed cost elements to communicate	7	link distributed cost elements
d	model selection and use depends on design resolution and information fidelity	13	define selection criteria for cost-evaluation tool / technology	12	provide guidance on tool / technology selection
e	individual model capabilities and requirements not systematically defined	12	define cost-evaluation tool / technology capabilities and requirements for use	11	provide access to and support the use of various cost evaluation tools / technologies
f	multiple models applied to component of design	15 16	identify relationships among models, tools/technologies define means for utilizing multiple estimates	13 14	support application of multiple models to design components aggregate results from multiple models and design components
g	cost models have numerous shortcomings when applied to conceptual and preliminary designs				

ref no.	Characteristics of cost evaluation environments	ref no	Needs to be met by, or through, the development of an IDCT Tool	ref no	Requirements for an effective IDCT Tool
11	***Data relevancy and currency***				
a	data widely distributed among numerous functional units throughout the organization	7	provide means for distributed cost elements to communicate	7	link distributed cost elements
b	data reside in many formats on a variety of platforms	6	provide means for disparate cost elements to communicate	6	link disparate cost elements
c	data requirements not explicitly defined	2 4	define user group roles and needs in developing and using cost estimates define processes for developing and using cost estimates	2 4	based on defined user group cost-evaluation roles and needs based on defined processes for developing and using estimates
12	***Islands of Technology***				
a	not well integrated; development and use in disparate, non-homogeneous environments, proprietary	3 4 5 6 7 20	identify interfaces between tools / technologies and user types define processes for developing and using cost estimates identify means for collaborative work provide means for disparate cost elements to communicate provide means for distributed cost elements to communicate identify sources and uses of knowledge	3 4 5 6 7 21	provide effective, intuitive interfaces for user groups based on defined processes for developing and using estimates function in a collaborative environment link disparate cost elements link distributed cost elements provide means to assimilate, synthesize, and aggregate cost models
b	lack of shared common capabilities and services across comparable domains, applications, technologies	12	define cost-evaluation tool / technology capabilities and requirements for use	11	provide access to and support the use of various cost evaluation tools / technologies
c	use/reuse not defined, how and when to use technology not defined	2 4 12	define user group roles and needs in developing and using cost estimates define processes for developing and using cost estimates define cost-evaluation tool / technology capabilities and requirements for use	2 4 11	based on defined user group cost-evaluation roles and needs based on defined processes for developing and using estimates provide access to and support the use of various cost evaluation tools / technologies

ref no.	**Characteristics** **of cost evaluation environments**	**ref no**	**Needs** **to be met by, or through, the development of an IDCT Tool**	**ref no**	**Requirements** **for an effective IDCT Tool**
d	development and maintenance ad hoc	2	define user group roles and needs in developing and using cost estimates	2	based on defined user group cost-evaluation roles and needs
		18	define means to manage changing cost elements	16	manage cost element change

Table 4-5 provides a summary of the requirements that were derived from mapping the characteristics of cost-evaluation environments to user needs and user needs to requirements in Table 4-4.

Table 4-5: Summary of derived requirements for IDCT

1	utilize dictionaries, common structures and hierarchies (e.g. cost, materials, processes, etc.)
2	based on defined user group cost-evaluation roles and needs
3	provide effective, intuitive interfaces for user groups
4	based on defined processes for developing and using estimates
5	function in a collaborative environment
6	link disparate cost elements
7	link distributed cost elements
8	support varied cost evaluation approaches
9	provide guidance on approach selection
10	provide access to all required information
11	provide access to and support the use of various cost evaluation tools / technologies
12	provide guidance on tool / technology selection
13	support application of multiple models to design components
14	aggregate results from multiple models and design components
15	utilize product information systems, e.g. PDM, CAD, requirements management
16	manage cost element change
17	provide means to explicitly address uncertainty in estimates
18	provide means to facilitate sensitivity analyses and synthesis of multiple estimates
19	provide access to knowledge sources and means to process and use knowledge
20	based on defined design decision-making processes relevant to cost evaluation
21	provide means to assimilate, synthesize, and aggregate cost models

In general, the IDCT Tool is a design decision-support system (DSS). As shown in Figure 4-8, its development requirements result from six primary components: (1) data, (2) models, (3) the specific design / cost-evaluation processes upon which it is based, (4) interfaces, (5) the users that it is designed to support, and (6) the enabling technologies, referred to as supporting technologies. The term “model” is used in a very general sense to represent costing systems, tools, models, etc. , i.e. any technology or methodology that provides cost evaluation support to the design decision-making processes. Also, in the following context, design means one alternative solution to a problem and is composed of a hierarchy of components.

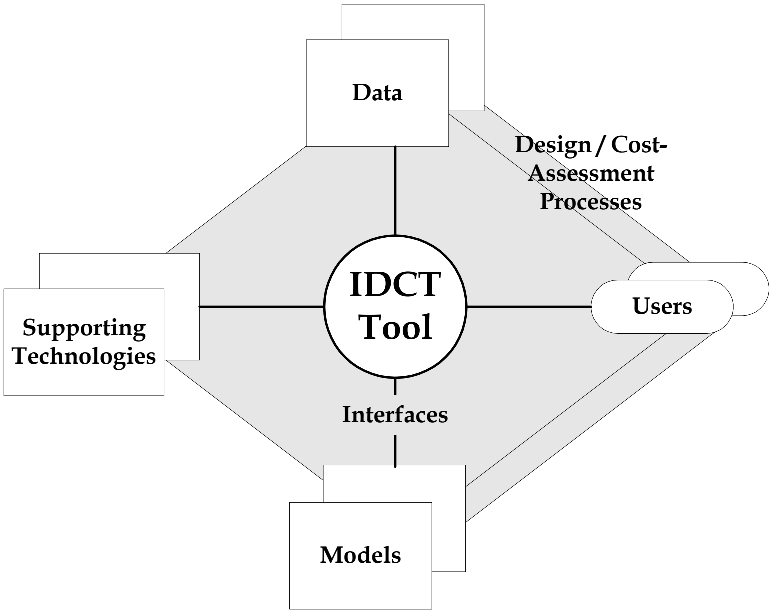


Figure 4-8: The IDCT Tool is a design decision-support tool.

The requirements are classified into the six DSS components below. The requirements from Table 4-5 are identified with an "R number," e.g. R-1 is "utilize dictionaries, common structures …," the first requirement in Table 4-5

1) **Data**: The IDCT Tool must manage two primary types of data: (1) all relevant data associated with the design that is being considered and (2) historical data from comparable systems/designs that are used in the cost evaluation processes.
   a) link disparate cost elements (R-6)
   b) link distributed cost elements (R-7)
   c) provide access to all required information (R-10), e.g. provide access to engineering, production, and operations parameters or characteristics for the design (and its components) that is being considered, engineering, production, and operations characteristics for items comparable to those being designed, existing production and operations cost databases for items comparable to the ones being designed, and support the investigation of the databases through access to exploration and analysis tools, e.g. for queries, statistical modeling, etc.
   d) manage cost element change (R-16), i.e., manage the cost estimates for the current design and alternatives designs that are under consideration and manage the cost estimates of the design over the life cycle.

2) **Models**: For any cost analysis, the IDCT Tool should use the most current and the most appropriate model(s) based on the information available. The Tool provides resources to all registered users. With minimal effort, all relevant models are available to be used to support the design decision-making processes.
   a) link disparate cost elements (R-6), i.e., models will function in a distributed, heterogeneous application environment
   b) link distributed cost elements (R-7), i.e., location of models is platform independent.
   c) provide access to and support the use of various cost evaluation tools / technologies (R-11); once models are "registered" with the system, they are available to all users. The registration process is

used to obtain information on the requirements of the model, its location, etc, but not how it works. The process should impose minimal requirements on the model owners.

d) manage cost element change (R-16)
   i) .In order to ensure the models are current, installation and maintenance of the models are the responsibility of their owners.
   ii) Model implementation details should be hidden; i.e. all that a general user needs know is what information the model provides and what information it requires, but is not concerned with how it makes that translation. This directly addresses the model maintenance concerns of users and the proprietary concerns of model owners.
   iii) The IDCT Tool should be useable by other systems, e.g. as input to a multidisciplinary design optimization system.
e) provide guidance on tool / technology selection (R-12) Model selection is based on the current design state.
   i) During the design process, model application is dynamic. The model that is used to assess the cost of a design depends on the information that is currently available. The cost model changes as the design changes and matures.
   ii) Heterogeneous models may be used within a design. Due to varying degrees of maturity of the components of a design, the more mature components' estimates should be based on actual production or operations data or from detailed cost models; whereas, newer and less-defined components, i.e. those with less design resolution, should employ high-level parametric tools.
f) support application of multiple models to design components (R-13) Multiple estimates can be made for a design or component. Estimates for the same design may be obtained from multiple models; this is one way to address uncertainty.
g) aggregate results from multiple models and design components (R-14) Multiple models may be applied to obtain a single estimate. Often a basic cost model is applied to a design, but must be adjusted to account for differences in quantity, time, etc. through the application of learning curve models, inflation adjustment models, etc.

3) **<u>Design / Cost-Evaluation Processes</u>**:
   a) utilize dictionaries, common structures and hierarchies, e.g. cost, materials, processes, etc. (R-1)
   b) based on defined processes for developing and using estimates (R-4)
   c) support varied cost evaluation approaches (R-8)
   d) provide guidance on approach selection (R-9)
   e) provide means to explicitly address uncertainty in estimates (R-17)
   f) based on defined design decision-making processes relevant to cost evaluation (R-20)

4) **<u>Interfaces</u>**: The IDCT Tool must support interactions with multiple user types, e.g. designers, cost engineers, managers, manufacturing engineers.
   a) provide effective, intuitive interfaces for user groups (R-3)
   b) link disparate cost elements (R-6)
   c) link distributed cost elements (R-7)
   d) provide access to all required information (R-10)
   e) provide access to and support the use of various cost evaluation tools / technologies (R-11)
   f) utilize product information systems, e.g. PDM, CAD, requirements management (R-15)
   g) provide access to knowledge sources and means to process and use knowledge (R-19)

5) **<u>Users</u>**: The IDCT Tool must support interactions with multiple user types, e.g. designers, cost engineers, managers, manufacturing engineers; i.e., it must:
   a) be based on defined user group cost-evaluation roles and needs (R-2), i.e., provide the resources required based on each user's role in a specific cost-evaluation process

b) provide effective, intuitive interfaces for user groups (R-3); provide separate views, interfaces, access, etc. depending on the user's needs
c) function in a collaborative environment (R-5)

6) **Supporting Technologies**: The IDCT Tool should utilize and leverage technologies that support cost evaluation processes.
a) function in a collaborative environment (R-5), e.g., collaborative electronic work environments
b) utilize product information systems, e.g. PDM, CAD, requirements management (R-15)
c) provide access to knowledge sources and means to process and use knowledge (R-19)
d) model selection, parameter specification, etc. are facilitated through machine intelligence and information technology

4.5.8 Identification of Potential COTS Software Tools/Technologies

The preliminary and high-level capabilities of an IDCT system were identified based on its system requirements and conceptual design. A search, mostly of the Internet, resulted in candidate software that could potentially be used to develop a comprehensive system for evaluating the cost of a design alternative as an integral part of the design process. Candidate COTS software was also identified at professional conferences and industry meetings. Table 4-6 summarize the commercial off-the-shelf (COTS) software that was found to be available for use in developing the IDCT system. The software in the table are grouped by major capability categories.

Detailed investigation into the software was beyond the scope of this project. For each of the software, the basic functionality and capabilities were identified. In many cases, demonstration or evaluation versions were reviewed. Many of the companies were contacted to solicit their participation in a follow-on project that would involve the actual development of an IDCT system. All of the companies that were contacted indicated a willingness to work with us. However, specific negotiations would occur under the follow-on project.

One of the first steps in a follow-on study would be to develop a prioritized list of development activities that are based on an industry survey and evaluation of the IDCT system. This list would identify the most important capabilities for an IDCT system and would provide the foundation for the selection of supporting technologies and their subsequent negotiations.

Table 4-6: Potential COTS software for an IDCT system.

Role	Name/Description	Contact information
Computer supported collaborative work	Stuffincommon	TeamWave Software Ltd. Web-Site: www.teamwave.com 100 DiscoveryPlaceOne 3553-31 Street N.W. Calgary, Alberta Contact: John K. Blood

Role	Name/Description	Contact information
	TeamWave Workplace	TeamWave Software Ltd. Web-Site: www.teamwave.com 100 DiscoveryPlaceOne 3553-31 Street N.W. Calgary, Alberta Contact: John K. Blood
	WebEx	WebEx Communications, Inc. Web-Site: www.webex.com 100 Rose Orchard Way San Jose, CA 95134
Cost models, tools and systems	ASCET (Aircraft System Cost Estimating Tool)	TECOLOTE Research, Inc. Web-Site: www.tecolote.com 5290 Overpass Road, Bldg. D Santa Barbara, California 93111-3011 Contact: Harmon Withee
	ACEIT (Automated Cost Estimating Integrated Tools)	TECOLOTE Research, Inc. Web-Site: www.tecolote.com 5290 Overpass Road, Bldg. D Santa Barbara, California 93111-3011 Contact: Harmon Withee
	Cost Advantage	Cognition Corporation Web-Site: http://www.ci.com/ Bedford, MA Contact: Mike Cronin
	COSTADE (Cost Optimization Software for Transport Aircraft Design Evaluation)	NASA Langley Research Center Hampton, VA W. T. Freeman, Jr. The Boeing Commercial Airplane Group Seattle, WA Larry Ilcewicz
	DeccanPro	Deccan Systems Inc. Web-Site: www.deccansystems.com 5935 Muncie Ct. Dublin, OH 43107-2601 Contact: Raja Mohan
	First Order Life Cycle Cost Model	ALE (Acquisition Logistics Engineering) Web-Site: www.ale.com 6797 North High Street, Suite 324 Worthington Ohio 43085 Contact: Charles O. Coogan
	Front End Analysis Trade Study Model	ALE (Acquisition Logistics Engineering) Web-Site: www.ale.com 6797 North High Street, Suite 324 Worthington Ohio 43085 Contact: Charles O. Coogan

Role	Name/Description	Contact information
	PRICE Systems Suite	PRICE Systems L.L.C Web-Site: www.pricesystems.com Suite 100 Wright Point 2 Office Building 5100 Springfield Pike Dayton, OH 45431
	Production Cost Model	Wallace and Company E-mail: dwsietman@msn.com 2940 Presidential Drive, Suite 390 Fairborn, OH 45324-6223
	SEER	Galorath Incorporated Web-Site: www.galorath.com 100 North Sepulveda Boulevard Suite 1801 El Segundo, CA 90245
	SmartCost	KBSI (Knowledge Base Systems Inc.) Web-Site: www.kbsi.com 1408 University Dr. East College Station, Texas Contact: Dr. Richard Mayer
Enabling Technologies	ICE (IDAPS Cost Estimation)	Frontier Technology Inc. Web-Site: www.fti-net.com 6785 Hollister Avenue Goleta, CA 93117 Contact: Ron Shroder
	Knowledge Center	Cognition Corporation Web-Site: http://www.ci.com/ Bedford, MA Contact: Mike Cromin
	ORB-IT	Systran Federal Corporation Web-Site: www.systranfederal.com 4027 Colonel Glen Hwy, Suite 210 Dayton, OH 45431 Contact: Dr. V. ("Nagu") Nagarajan
	Orbix	IONA Technologies PLC Web-Site: www.iona.com The IONA Building Shelbourne Road Dublin 4
	OZ (Object czar)	Knowledge Base Engineering Web-Site: www.kbe.net 71 Rhoads Center Drive Centerville, OH 45458 Contact: Sam Nusinow
	StepTools	STEP Tools, Inc. Web-Site: www.steptools.com 216 River Street Troy, New York 12180
	VisiBroker	Inprise Corporation (Division of Borland) Web-Site: www.borland.com 100 Enterprise Way Scotts Valley CA, 95066

Role	Name/Description	Contact information
	Rational Suite DevelopmentStudio	Rational Software Corporation Web-Site:www.rational.com 18880 Homestead Rd. Cupertino, CA 95014 Contact: Kim Boehm
Materials costing and properties databases	CenBASE	Centor Software Corporation Web-Site: www.centor.com 2171 Campus Drive, Suite 260 USA
	Concept Computer Systems Product Costing – Material Databases	Concept Computer Systems Ltd Web-Site: www.conceptsystems.com 3-4 Alexandra Terrace, Alexandra Road Aldershot, Hants, GU11 3HU, England
	DeccanPro	Deccan Systems Inc. Web-Site: www.deccansystems.com 5935 Muncie Ct. Dublin, OH 43107-2601 Contact: Raja Mohan
	Matches Cost Software	Matches Inc. Web-Site:www.matche.com 2005 Mistletoe Lane, Edmond OK 73034-6054 Contact: David A. Miligan
	MPDB (Materials Properties Database)	JAHM Software, Inc. Web-Site:www.jahm.com 103 Hill Street, Suite 9 Stoneham, Massachusetts 02180-3710
Process modeling	ABC FlowCharter	Micrografx Inc. Web-Site:www.micrografx.com 8144 Walnut Hill Ln. Suite 1050 Dallas, TX 75231
	AI0 WIN	KBSI (Knowledge Base Systems Inc.) Web-Site: www.kbsi.com 1408 University Dr. East College Station, Texas Contact: Dr. Richard Mayer
Reliability assessment	Relex	Relex Software Corporation Web-Site: www.relexsoftware.com 540 Pellis Road, Greensburg, PA 15601 USA Contact: Vince Elias
	Reliability Workbench	Isograph, Inc. Web-Site: www.isograph.com Continental Grand Plaza II 400 Continental Blvd., 64th Floor El Segando, CA 90245
	ITEM	Item Software Web-Site:www.itemsoft.com 2190 Towne Centre Place, Suite 314 Anaheim, CA 92806
Requirements management	DOORS	Quality Systems & Software Web-Site: www.qssinc.com Northbrook House, Oxford Science Park Oxford, United Kingdom OX4 4GA

Role	Name/Description	Contact information
	SLATE	TD Technologies, Inc. Web-Site: www.tdtech.com 2425 N. Central Expressway, Suite 200 Richardson, TX 75080
	Rational RequisitePro	Rational Software Corporation Web-Site:www.rational.com 18880 Homestead Rd. Cupertino, CA 95014 Contact: Kim Boehm

4.5.9 Review of Reliability Prediction During Conceptual Design.

Since the IDCT system is to address life-cycle cost and since reliability is a major driver in life-cycle cost, a thorough review of the reliability literature was conducted to identify means for assessing reliability during conceptual design.
Reliability prediction is simply the analysis of parts and components in an effort to predict the rate at which an item will fail. Improvements in product design have been accomplished by reliability prediction for five decades; however, the focus has traditionally been on detailed design. Tests of prototypes and existing field data are used to estimate reliability performance. MIL-HDBK-217 and Bellcore models are usually used for reliability prediction during this stage of design.

Several information venues were utilized but the primary sources were the review of three journals: the *Annual Proceedings of the Reliability and Maintainability Symposium*, *IEEE Transactions on Reliability*, and *Naval Research Logistics*. The proceedings were reviewed from 1965 to the present and the two journals were reviewed from 1960. The EBSCO Host search engine resulted in one relevant journal articles, from the *Journal of Engineering*. The Internet was also utilized for information. Yahoo and Lycos were the search engines used for the research. The Relex Software Corporation web site (www.relexsoftware.com/predict.htm) contained a plethora of information relating to reliability prediction. This site was extremely helpful in gaining insight on popular reliability prediction tools like MIL-HDBK-217 and Bellcore.

The most widely known and used reliability prediction handbook is MIL-HDBK-217, the Military Handbook for “Reliability Prediction of Electronic Equipment”. It is published by the Department of Defense, based on work done by the Reliability Analysis Center and Rome Laboratory at Griffis AFB, NY. MIL-HDBK-217 contains failure rate models for the various part types used in electronic systems. It contains two methods for reliability prediction; the *part stress analysis* and the *parts count analysis*. These methods are utilized depending on the amount of information available for the components. The *part stress analysis* requires the greatest amount of information and is applicable during the latter part of the detailed design phase. To incorporate this method, a detailed parts list must be available with specific stress values for each component. The *parts count analysis* requires less information to estimate component failure rates. The information needed to conduct this analysis is:

- generic part type (resistor, capacitor, etc.)
- part quality levels
- equipment environment
- part stress and temperature levels

Using this information, the model produces an estimated failure rate using the formula below:

$$\lambda_p = \pi_Q \cdot \pi_L \cdot \left[C_1 \cdot \pi_T + (C_2 + C_3) \cdot \pi_E\right] / 10^6 \, hours$$

where:

λ_p = Part failure rate
π_Q = Quality factor
π_L = Learning factor
π_T = Temperature factor (Arrhenius Model)
π_E = Environment factor

and C_1 and C_2 are complexity factors based on chip complexity. C_3 is a packaging complexity factor.

This analysis can be applied during the latter stages of the conceptual design phase. However, the component must be comprised of all generic parts that are included in the Handbook's model.

MIL-HDBK-217 quickly became the standard for reliability prediction of electronic systems. However, several industries such as the telecommunications industry, needed specific reliability prediction tools that MIL-HDBK-217 could not perform. Corporations developed their own prediction models; the most well known is the Bellcore Reliability model developed at AT&T Bell Labs with a focus on telecommunications. The main concepts between MIL-HDBK-217 and Bellcore are very similar, but Bellcore added the ability to take into account burn-in, field, and laboratory testing. Bellcore uses part stress and part count analysis techniques.

The Handbook of Reliability Prediction Procedures for Mechanical Equipment (NSWC-94/L07), developed by the US Navy, provides a model to predict reliability for mechanical devices including springs, bearings, seals, motors, brakes, and clutches. It is relatively new and not as widely accepted as MIL-HDBK-217 and Bellcore.

As this search revealed, the ability to predict reliability during the early stages of conceptual design is very limited. Without a prototype to test in a lab environment or field data, stresses and failure rates are usually unknown. A popular method for early reliability prediction is to develop a computer model for the system. However, most of these models are extremely specific to an individual system or industry.

A generic simulation model has been developed for predicting reliability without knowing the specific failure rate for the components in the system. The system structure for this model is a series of non-repairable subsystems with active or stand-by redundancy within each subsystem. The failure rates for the system's components are constant. Component failure rates that are unknown are modeled using the triangular distribution. The mission profile for the system is a single mission of finite length. The model estimates mission reliability, average time to failure for the system, system and subsystem life distributions, and mission cost. The estimates are based on the number of series subsystems and active or stand-by components for each subsystem, known component failure rates, distribution parameters for components with unknown failure rates, mission length, and component and mission costs. Potential future improvements to this model include allowing repairable subsystems and linking information directly to MIL-HDBK-217.

In conclusion, most reliability prediction tools are not applicable during the conceptual design phase. Monte-Carlo simulation can be used to obtain failure rate estimates; however, these estimates are quite rough and require a lot of statistical evaluation.

4.5.10 Master Plan for the Next Phase of Work

The anticipated tasks for a follow-on project that would continue the development of the IDCT system are outlined below. The approach for developing the IDCT Tool is presented in terms of three main interrelated aspects: (1) design/cost-evaluation processes, (2) concept of operation (ConOp), and (3) design and development of the IDCT Tool. The first aspect is tied to the need to understand the design and cost-evaluation processes, as well as the relationships between the processes. The second aspect focuses on the users of the tool and understanding how the tool will be inserted and used during design. The final aspect involves the design and development of a tool that is based on established processes and clear needs and is validated by industry stakeholders.

The development should be highly iterative. For example, as the tool is designed, developed, and tested, it should be evaluated on how well it supports the design/cost-evaluation processes. Conversely, tool design and testing activities should lead to better understanding of the design/cost-evaluation processes and will therefore result in a refined definition of the processes. The tool development should spiral from the general conceptual design to specific detailed implementations, but should always be guided by the overall objective, understanding of the underlying processes, and concept of operations. The customer and a technical review board (TRB) should guide the project activities.

This approach is analogous to Rummler-Brache's three-level framework for improving performance (Rummler, 1995). The first or Organization level provides the macro-level context or systems view and is used to identify major functions. The second or Process level is a cross-functional workflow of how work gets done – in this case, how cost evaluations are performed during design. The third or Job/Performer level focuses on the people doing their work and the tasks. Since this project focused on conceptual design, little has been done at this level; therefore, it will be a major component of follow-on study. It will provide the bases for the design and development of the IDCT Tool and its insertion into the organization. This framework, as well as tools and processes associated with it, is credited with providing a means for: diagnosing and eliminating deficient performance, an engine for continuously improving systems that are performing adequately, a road map for guiding new directions, and a blueprint for designing new entities. Similar outcomes are expected through the application of the framework to develop an IDCT Tool.

The first two aspects would each have two primary activities, research/definition and review. The design/cost-evaluation processes' aspect is the foundation of the IDCT Tool; it defines *what* the tool must do. The ConOp specifies *how* the tool must function when it is used. Both of these, while defined in this study, would be iteratively defined in more detail as the system evolves. As the processes in which the tool will operate and the specific jobs it must perform become better understood through interaction with industry, subsequent versions of the prototype of the tool would be enhanced, tested, and evaluated. The third aspect combines both the design and the development of the IDCT Tool; it also includes research/definition and review activities that support and evaluate design and development efforts.

The tasks for a follow-on project that develops the IDCT Tool are outlined below. The tasks are grouped into four major task areas: Direct Extensions of the Current Project, Design/Cost-Evaluation Processes, Concept of Operations, and IDCT Tool. The timing of the tasks and the relationships among the tasks are shown in the project workflow, Figure 4-9, at the end of this section.

1. Direct Extensions of the Current Project
   1.1. Formally evaluate and validate the work from the current project.
      - Develop a survey instrument.

- Distribute survey to industry, government, and university personnel (via email). The evaluators will be directed to the project website which will contain the current version of the IDCT Tool prototype and relevant project documents..

1.2. Analyze system capability

- Analyze and validate the preliminary object-based hierarchical IDCT constructs, including basic item structure, product form, material, manufacturing process, and cost structure. This will be accomplished via the aforementioned survey.
- Investigate model management methodologies, i.e. identify means to represent, store, maintain, and utilize models in design trade study processes.
- Reaffirm the functionality required to create, use, and control design trade studies. This will be accomplished via the aforementioned survey.
- Ensure cost instances can be recorded and tracked. These instances must contain design characteristics, models used, resulting measures of design performance, as well as designer rationale for design changes.
- Ensure cost instances defined above can be "mined" for analysis and comparison to similar designs and for calibrating models.

1.3. Develop team

- Develop specific plans for each team member in order to validate their roles and responsibilities.
- Provide plans for future development to project sponsors and representatives from the design community.
- Solicit additional participants via the aforementioned survey.

1.4. Project support

- Develop and maintain (on the web) a dictionary of standard design/cost trades terminology in order to utilize and integrate disparate cost models. For example, as models are added to the model base, each input and output would be linked to the terms in the dictionary, either directly or through the use of synonyms or aliases. Variables that do not fit existing definitions would be added to the dictionary at the appropriate place in the taxonomy.
- Identify an illustrative design case study that will be used to evaluate available affordability/cost assessment technologies and demonstrate the application of the IDCT Tool. The design case will include components with varying degrees of technological maturity in order to demonstrate the application and assimilation of multiple affordability/cost assessment technologies and should include Make-or-Buy decision processes.

1.5. Create a project website

- Develop a strategic plan for use of the website. It is anticipated that the website will be a primary vehicle for team communication, outreach, and feedback/evaluations. We also envision the website being the primary source of information and links on design/cost-evaluation trade studies.
- Identify the type of information that will be available through the website, as well as means to access the information, tracking, maintenance, etc.
- Announce the website via a mass emailing; solicit relevant sites to provide links to the project site.

2. Design/Cost Evaluation Processes

- The purpose of these activities is to validate and extend the definitions of the Design/Cost-Evaluation processes from the present project. It is imperative that these processes are well defined since they are the bases for the development of the IDCT Tool. Process definition provides excellent opportunities for introducing and involving potential users of the IDCT Tool in the development effort, as well as involving technology providers whose products may be a part of the completed tool.

- While this will be an ongoing effort throughout the project, the focus will be at the front-end of the project.
- A primary source of information for defining and modeling the processes will be through field studies that are described below.

2.1. Define design processes

- Define design-related processes and activities to the level of detail necessary to identify functions and requirements for the IDCT Tool and in sufficient detail to identify means for integrating the Tool into the processes.
- Model the processes and activities in an IDEF or similar methodology.
- Identify a means to associate/link/incorporate the process/activity definitions with the IDCT Tool.

2.2. Define cost-evaluation processes

- Define the cost-evaluation processes and activities that are relevant to the early design phase to the level of detail necessary to identify functions and requirements for the IDCT Tool and in sufficient detail to identify means for integrating the Tool into the processes.
- Model the processes and activities in an IDEF or similar methodology.
- Identify a means to associate/link/incorporate the process/activity definitions with the IDCT Tool.

2.3. Define relationships and interfaces between the design and cost-evaluation processes.

- The focus of these activities will be on defining:
- key interfaces between the design and cost evaluation processes; most opportunities for improvement are at the interfaces between processes.
- temporal flow of activities in order to address the dynamic effects of transitions and transformations in activities, models, data, needs, etc. as the design evolves and matures,
- feedback mechanism to the different stakeholders, in terms of the type of information, format, media, frequency, etc. – i.e., identify what information is needed by whom, when, and in what form in order to enhance their decision-making capabilities, and
- use characteristics and services that would encourage and facilitate the utilization of the IDCT Tool by the stakeholders.
- Integrate/map the design and cost-evaluation process models in order to identify the key interfaces between them.
- Define the interfaces to the level of detail necessary to identify functions and requirements for the IDCT Tool
- Based on the functional analyses and requirements definition, identify and define the services that the Tool must provide for cost-evaluation to effectively support design decisions. These services form the basis of the IDCT Tool development efforts. General service classes include: decision making, presentation, communication, data storage and retrieval, hierarchy management, model management.

2.4. Process definition data collection and evaluation

2.4.1. Industry surveys

- Develop a means to collect design/cost-evaluation process information as part of the precursor work validation survey described above.
- Develop and administer a means to evaluate our representation of the design/cost-evaluation processes; e.g. website survey, presentation feedback survey.

2.4.2. Field studies – a primary means to collect and validate information on design/cost-evaluation processes is through industry field studies.

- Select three companies to participate in the design/cost-evaluation process definition field studies.

    - Develop a set of activities to be conducted during the site visit, e.g. interviews, demonstrations, experiments, process mapping exercises, data collection needs.
    - Develop pre-visit surveys to aid in site visit planning, gather as much information as possible before the site visit, and identify the appropriate people to interview, data to collect, etc.
    - Conduct on-site activities over an approximately one-week time period (for each company).
    - Document the activities and results of each field study and have the company review its contents.
    - Assimilate the information obtained from the field studies.
    - Report the findings to the TRB and release the report through the project website. Solicit more widespread feedback through the website.
    - Utilize the process definitions and other findings from the field study to develop a prioritized set of IDCT Tool requirements.

  2.5. Process definition management
  - Baseline the process definitions from precursor project.
  - Develop a change management process for updating the design/cost-evaluation process definitions based on data collection activities. Changes will be reviewed by the TRB before release.

  2.6. Reviews
  - In addition to providing definition and a means for data collection, the industry surveys that were defined above will provide feedback and a formal means of review and evaluation of the processes that will utilize IDCT tool.
  - Conference presentations and demonstration will also be an important source of feedback on our definition and representation of the design/cost-evaluation processes.
  - At the conclusion of the field study portion of the project, as described above, industry participants will formally review the assimilated process definitions.
  - The TRB will review and evaluate the design/cost-evaluation process definitions. Since these processes are a primary component of the project, changes to the defined processes will be reviewed at each TRB meeting.

3. Concept of Operations
- Since the concept of operations is tightly linked to the processes definitions, similar activities will be performed in order to adequately understand and define how the IDCT Tool will be utilized in practice.
- The purpose of the following activities is to validate and extend the concept of operations that was developed in the precursor study. It is imperative that the use of the IDCT Tool be fully understood and clearly defined since it is a primary basis for the development of the IDCT Tool.
- The concepts of operations will be explored from two perspectives: (1) users or consumers of information provided by the IDCT Tool (e.g. designers and IPTs) and (2) producers of information, models, etc. that are incorporated within the Tool (e.g. domain experts, information technology professional)
- While this will be an ongoing effort throughout the project, the focus will be at the front-end of the project.

3.1. Define the concepts of operations to the level of detail necessary to identify functions and requirements for the IDCT Tool and in sufficient detail to identify means for integrating the Tool into the processes.

3.2. Link the concepts of operations to the design/cost evaluation processes.

3.3. Based on the functional analyses and requirements definition, identify and define the services that the Tool must provide in order to effectively support design decisions.

- These services form the basis of the IDCT Tool development efforts. General service classes include: decision making, presentation, communication, data storage and retrieval, hierarchy management, model management.
- The precursor study provides a list of anticipated services; however this research activity is aimed at defining specific services.

3.4. Process definition data collection, evaluation

3.4.1. Industry surveys

- Develop a means to collect, characterize, and represent the concepts of operations.
- Develop and administer a means to evaluate our representation of the concepts of operations; e.g. website survey, presentation feedback survey.

3.4.2. Field studies – a primary means to collect and validate information on concepts of operations is through industry field studies. (This is the same as described above for defining design/cost-evaluation processes; i.e., concepts of operations will be investigated concurrently with the process definitions.)

- Select three companies to participate in the field studies.
- Develop a set of activities to be conducted during the site visit, e.g. interviews, demonstrations, experiments, process mapping exercises, data collection needs.
- Develop pre-visit surveys to aid in site visit planning, gather as much information as possible before the site visit, and identify the appropriate people to interview, data to collect, etc.
- Conduct on-site activities over an approximately one-week time period (for each company).
- Document the activities and results of each field study and have the company review its contents.
- Assimilate the information obtained from the field studies.
- Report the findings to the TRB meeting and release the report through the project website. Solicit more widespread feedback through the website.
- Utilize the concepts of operations and other findings from the field study to develop a prioritized set of IDCT Tool requirements.

3.5. Concepts of operations management

- Baseline the concepts of operations definitions from precursor project.
- Develop a change management process for updating the concepts of operations based on data collection activities. Changes will be reviewed by the TRB before release.

3.6. Reviews

- In addition to providing definition and a means for data collection, the industry surveys that were defined above will provide feedback and a formal means of review and evaluation of the how the IDCT tool will be utilized.
- Conference presentations and demonstration will also be an important source of feedback on our definition and representation of the use of the IDCT Tool.
- At the conclusion of the field study portion of the project, as described above, industry participants will formally review the assimilated definitions.
- The TRB will review and evaluate the concepts of operations definitions. Since they are a primary component of the project, changes will be reviewed at each TRB meeting.

4. IDCT Tool

4.1. Research/Definition

4.1.1. Refine the methodology for characterizing, describing, and classifying cost-evaluation technologies (CETs).

- A CET is a software technology that provides affordability/cost evaluation functionality, e.g., ACEIT, AFTOC, Cost Advantage, PRICE, SEER.
- The taxonomy characterizes existing cost technologies in terms of output provided (cost elements), input required, applicable product/process domain(s), applicable phase(s) of product life cycle, evaluation methodologies utilized, resource requirements, applicable domains, etc.
- The taxonomy provides the foundation for model and methodology selection, a primary service of the IDCT tool.
- Review the appropriateness of the classification as each CET is investigated and utilized by the IDCT Tool. Update the methodology as needed.

4.1.2. Investigate <u>each</u> cost-evaluation technology that will be utilized by the IDCT Tool.

- Install in MSU's Advanced Design/Cost Technology Lab.
- Review the documentation, including users' guides, tutorials, example applications, articles and papers, etc.
- Receive training, as required, from the technology provider.
- Apply the technology to the design case study.
- Map the cost and related measures of system performance that is provided by the technology into the measures hierarchical classification.
- Map the inputs required by the technology into the input hierarchical classification.
- Prepare a report on the technology and its role in the IDCT.
- Develop a list of implementation issues that must be addressed in order for the IDCT Tool to effectively utilize the technology. Develop a plan for addressing the issues.

4.1.3. Refine the methodology for characterizing, describing, and classifying development support technologies (DSTs) in terms of the support services (functions and capabilities) that they provide and the IDCT needs that they meet.

- DSTs are enablers or software that enhances the functionality of a CET, e.g., ICE, I-DEAS, Knowledge Center, MetaPhase, ORB-IT, OZ.
- Example support services/technologies include those that:

4.1.3.1. *relate* and *integrate* existing and emerging cost evaluation technologies,

4.1.3.2. *manage* and *control* the design/cost evaluation processes and the interactions of those processes with CETs, both individually and collectively, e.g. model and data retrieval, selection, storage and warehousing, versioning controls and rationale documentation, maintenance, and

4.1.3.3. enhance the design decision-making processes through the *use* of CETS, e.g. links to other modeling and analysis tools, including simulation and optimization, utilization of collaboration and communications technologies, links to specific information sources and references, application of graphics and visualization technologies, provisions for learning from design experiences, and generation of reports and documentation.

- Baseline the support services required by the IDCT that were identified in the precursor study and the methodology for describing and classifying the services and their associated technologies that provide the services. Information (reviews, papers, web links, etc.) on candidate technologies were collected and catalogued in the precursor project.
- Review the appropriateness of the classification as each DST is investigated and utilized by the IDCT Tool. Update the methodology as needed.

4.1.4. Investigate <u>each</u> DST that will be utilized by the IDCT Tool.

- Install in MSU's Advanced Design/Cost Technology Lab.
- Review the documentation, including users' guides, tutorials, example applications, articles and papers, etc.

    - Receive training, as required, from the technology provider.
    - Prototype the application of the technology to the IDCT Tool.
    - Prepare a report on the technology and its role in the IDCT.
    - Develop a list of implementation issues that must be addressed in order for the IDCT Tool to effectively utilize the technology. Develop a plan for addressing the issues.

4.2. Design

- Part of the analysis of precursor work will be to validate the conceptual design. As discussed above, this will be accomplished through industry surveys, TRB review, and presentations at various professional conferences.
- Part of the analysis of precursor work will be to develop a prioritized preliminary design plan. This plan will be used to direct development efforts until the detail design is formulated.
- Formulation of the detailed design will begin once the preliminary design is approved. It will be formulated as the preliminary design is being developed and constructed. While the detailed design will be based on inputs from a variety of sources (e.g. reviews, TRB), the primary driver will be the results of design/cost-evaluation and ConOp field studies.

4.3. Development

- As described above, the IDCT Tool will be "fast tracked," i.e., design and development will activities will have considerable overlap.
- Continued development of the IDCT Tool, based on the prototype from the precursor project, will begin with the acceptance of the conceptual design and preliminary design plan. Therefore, there are two development steps, one based on the preliminary design and one based on the detailed design.

4.3.1. Construct a plan that includes a prioritized set of activities for developing the IDCT.

- The plan will address the allocation of requirements to work packages, sequencing of the design and development activities, procedures for modular testing during development, procedures for periodic "system" testing, etc.
- Allocate specifications to work packages so that the Tool can be developed in modules. The modular allocation will facilitate the integration of multiple affordability/cost evaluation technologies as well as parallel development activities.
- Priorities will be based on the criticality of the activity to the overall capability of the IDCT, risk of not completely completing the activity, relationship to other development activities, etc.
- The primary types of activities that will be considered include (1) integration of technologies, (2) management and control of processes and technology components (e.g. data, models), and (3) facilitation and enhancement of the utilization of affordability/cost evaluation trade study tools early in design (e.g. ease of use, timeliness of response, quality of assessment).
    - An example of the first type of activity is an investigation of the integration of product design data (from CAD/CAM/CAE systems) and primary costing systems. For example, investigating using SDRC's (Structural Dynamics Research Corporation) product data management (PDM) system and system engineering tools (MetaPhase) to supply information to a primary parametric cost estimating tool (e.g. SEER-H from Galorath, Inc.) through the Integrated Desktop Analysis and Planning System (IDAPS) Cost Estimation (ICE) from Frontier Technology.
    - An example of the second type of activity is model/data selection – matching a set of product/process design parameters and the desired cost elements with a cost-evaluation model(s) that utilizes the parameters and provides the requisite

elements. An associated activity is handling selection when such models cannot be identified or if several models are close to meeting the criteria. In either case, the variables that would need to be specified in order to allow the model to be utilized would be communicated to the user.

- An example of the third type of activity would be to define and develop a means to track/record cost-evaluation instances. Each time a design is evaluated in terms of cost, i.e. the cost–evaluation process is invoked, characteristics of the design, model(s) used, and resulting metrics are recorded. This would provide a rich history of the design's evolution and rationale for the changes. It would also provide an extensive database that can be mined for the purpose of benchmarking against similar designs/products and calibrating models. It could also be used as a pre-processor to search for feasible initial designs that have been previously considered.

4.3.2. Develop procedures for testing modules as well as the entire system.

- Identify the goals of each test and how they will be measured.
- Formulate a detailed test and evaluation plan for universities and industry, including frequency, responsibilities, reporting, etc.; present the plan at the preliminary design review.
- Obtain permission from affordability/cost evaluation technology vendors for use of their software during the tests.

4.3.3. Perform the development activities that are outlined in the preliminary design implementation plan.

4.3.4. Perform the development activities that are outlined in the detailed design implementation plan.

4.3.5. Develop IDCT Tool documentation as the system is designed and developed. Provide draft manuals as the project evolves, the final set will be provided at the conclusion of the project.

4.3.6. Test and evaluate the IDCT Tool

- Evaluate the results of the test and take corrective action.
- Document the evaluation and resulting actions.

4.3.7. Upon completion of the contract, deliver both run time and source code versions of the developed software.

4.4. Reviews

- The design and development of the IDCT will be under nearly constant review. As described earlier there are numerous opportunities to obtain feedback and guidance on the design and development of the IDCT Tool through, both formal and informal means.
- Evaluation instruments will be developed for each type of review so that comments are appropriately documented.

4.4.1. Conceptual Design Review

- The conceptual design for the IDCT Tool is an outcome of the precursor project. It will be evaluated as part of the "Analysis of Precursor Work" activities.
- The TRB will formally review the conceptual design prior to the first meeting. The preliminary design plan will be presented at that meeting.

4.4.2. Preliminary Design Review

- The preliminary design plan will be reviewed at the first TRB meeting.
- Development using the preliminary design of the IDCT Tool will evolve between the time when the results of the preliminary design are reviewed and the detailed design plan is presented.

- An interim review of the development of the IDCT Tool using the preliminary design will be held at a TRB meeting.

4.4.3. Detailed Design Review

- The detailed design plan will be reviewed at a TRB meeting.
- The detailed design of the IDCT Tool will evolve between the time the detailed design plan is presented and when the Beta test plans are presented.
- An interim detailed design review will be held at a TRB meeting.

4.4.4. Student reviews and evaluation

- Develop an evaluation survey instrument to collect feedback from the students using the IDCT Tool as part of their design course.
- Conduct student evaluations of the Tool; document, analyze, and report the TRB.

4.4.5. Beta-site test and evaluation of the IDCT Tool.

- Identify the goals of each test and how they will be measured. While testing will be encouraged at as many sites as feasible, there will be two primary test sites. The budget includes two trips to the two primary sites, one early in the testing process for installation and training and the other near the end of the test period for a review. The final presentation will be held at the last test site.
- Identify and obtain commitments from Beta test sites.
- Identify and define the test articles that are to be used during testing.
- Develop a test and evaluation (T&E) plan including frequency, responsibilities, reporting, etc.
- Develop a set of test and evaluation procedures that define how the tests will be conducted, evaluated, and documented.
- Conduct Beta site tests.
  - Install IDCT Tool and supporting technologies.
  - Train, as necessary, Beta site testing personnel.
  - Monitor activities according to the T&E plan.
- Evaluate and document the results of the test.
- Take corrective action or develop a plan for addressing the issues raised during the test.
- Document the testing process and results.

4.4.6. Informal reviews

- Informal reviews will be obtained from conference presentations, field studies, and the website.
  - Develop a review and evaluation plan.
  - Develop a methodology for assimilating and analyzing the various review instruments.
  - Develop an evaluation survey instrument to collect feedback at conference presentations on the design and development of the IDCT Tool.
  - Develop an evaluation survey instrument to collect feedback on the design and development of the IDCT Tool during the field studies.
  - Develop an evaluation survey instrument to collect feedback on the design and development of the IDCT Tool from the project website.
  - Conduct reviews at conference presentations, field studies, etc.
  - Document, analyze, and report outcomes from the surveys and reviews.

4.5. Reporting and dissemination of results.

- Publish on the project website.
- Prepare a quarterly newsletter that describes the project's activities and progress. The newsletter will be hosted at the website; notifications of the postings will be via email.

- Develop a final report.

Figure 4-9 outlines how the tasks that are associated with the statement of work are to be carried out over the duration of the project, in terms of workflow.

	Quarters after project start													
MILESTONES	1	2	3	4	5	6	7	8	9	10	11	12	13	14
Project kickoff	Δ													
Develop, administer, and analyze validation survey.		Δ--Δ	---Δ											
Rollout website.			Δ											
Technical Review Board: Conceptual design review; Preliminary design plan review.			Δ											
Technical Review Board meeting: Interim review of preliminary design development; Field study plan review					Δ									
Design/Cost-evaluation Field Studies					Δ----	---Δ								
Technical Review Board: Review of development under preliminary design; Detailed design plan review; Results of Field Study.							Δ							
Test the application of the IDCT Tool in a design course at MSU.								Δ--	-Δ					
Technical Review Board: Interim review of detailed design development; Technology field study plan review; Beta test plan review									Δ					
Technology Field Studies									Δ----	----Δ				
Technical Review Board: Review of detailed design development; Results of Technology Field Study; Beta test plan review.											Δ			
Beta Test at site #1											Δ----	Δ		
Beta test at site #2												Δ---	-Δ	
final Technical Review Board													Δ	
Final report complete; end of project.														Δ

Figure 4-9: Major milestones and workflow for next phase of work.

4.5.11 Conceptual Foundation for an IDCT Tool

Having described the characteristics of the cost-evaluation environment and having identified the requirements for effectively supporting the design process, we now present the conceptual foundation for an IDCT Tool.

Figure 4-10 provides a visual representation of the general concept for an IDCT Tool. It is based on the notion that cost evaluation transforms design variable values into values for specified cost measures. The feedback loop at the bottom of the figure illustrates the use of the cost measures by the designer and/or IPT to change the value of the design variables in order to improve the cost performance of the design. Each of the components of the process that are represented in the figure is further discussed below.

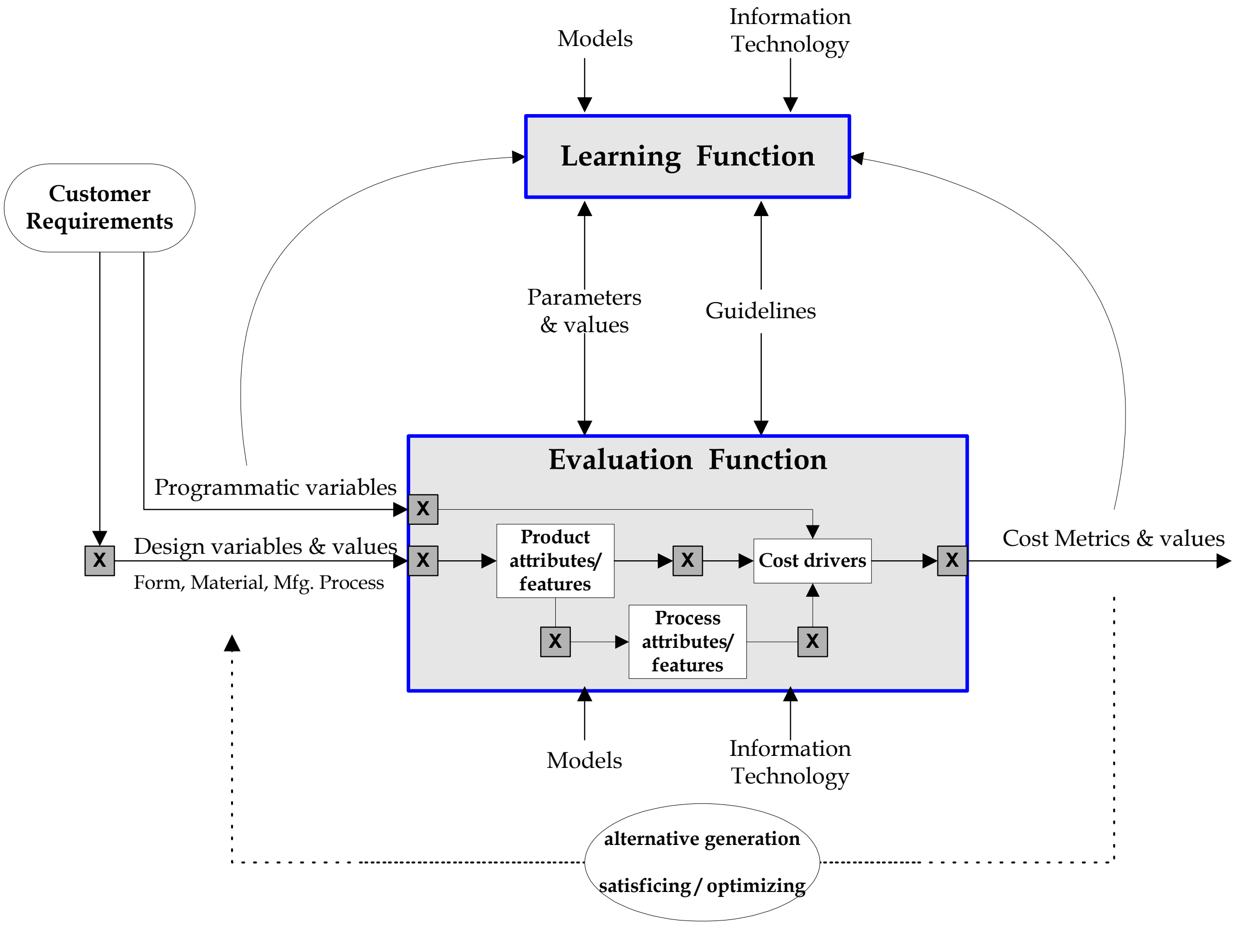


Figure 4-10: The conceptual foundation for an IDCT Tool.

In keeping with the functions defined earlier, and as shown in Figure 4-10, cost evaluation fundamentally involves *evaluation* and *learning*. Evaluation includes the basic cost estimating methodologies; it typically operates in a feed-forward mode, from specifying values for design variables to obtaining cost performance measures. The evaluation function is mostly utilized during the “using” cost-evaluation process that was defined earlier. The learning function typically operates in a feed-backward mode in that knowledge is obtained mostly in the form of guidelines, from design experiences. The learning function provides much of the basis for the “producing” cost-evaluation process that was defined earlier. In order to provide a basis for learning, each design evaluation needs to be captured as an instance or transaction and then periodically processed in search of guidelines for use in future designs. Such guidelines obviously help avoid repeating mistakes, provide suggestions that shorten design time, and provide better default values for variables (e.g. assuming a manufacturing process based on “similar” products.)

Enabling information technologies are included within the conceptual foundation since they are essential to realize, manage, and coordinate the components. Examples of such technologies include: database and knowledge base management systems, user and system interfaces, integration and communication tools (e.g. internet, CORBA), search engines, computational and display support, etc. As shown in Figure 4-10, information technologies provide support to both the evaluation and learning functions.

Similarly, models play a major role in cost evaluation. As shown in Figure 4-10, they support both the evaluation and learning functions. Generally, models are used by the evaluation function and produced or maintained through the learning function. In order to carry out the evaluation and learning functions, the IDCT Tool must provide for certain type of behaviors; i.e., be able to perform a variety of *actions*. For example, in some way the appropriate model(s) must be selected in order to be used to evaluate a design based on the amount and type of information available. Similarly, a means must be provided to select “similar” designs for comparing or contrasting.

4.5.11.1 Evaluation function

The evaluation function is broader than estimating. We view *estimating* as the simplest of three evaluation sub-functions in that it provides performance measure(s) for a single design. A second sub-function, *comparing*, involves the simultaneous consideration of multiple designs, with the focus on similarities. For example, prior to making certain decisions, a designer may investigate what worked well in the past on comparable designs. The third evaluation sub-function is *contrasting*. It also involves the simultaneous consideration of multiple designs, but with the focus on differences. There are two types of contrasting -- one involves *choice*, i.e. selecting from a set of alternatives based on differences; the second involves *tracking*, i.e., monitoring design performance over time to note changes and reasons for the changes.

As noted earlier, cost-evaluation, especially the evaluation function, is fundamentally a transformation process. The process is composed of a set of components. Inputs to the process include specific values for a variety of variables/parameters. These include decision or design variables, e.g. material type, shape, size, manufacturing processes used to produce the product. They also include supporting parameters -- parameters that describe elements that affect cost but are not directly under the control of the designer/IPPD team; e.g., labor rates, overhead rates, scrap rates, raw material price, and, general system parameters (quantity, schedule, year dollars, etc.). Output from the transformation process are values for criterion variables or *metrics* that provide a measure of the design’s performance, including relative and absolute indicators of cost, manufacturability, etc. Models are key components in the process since they transform variables that characterize the product into performance metrics; they include all models, functions, algorithms, etc. that support an evaluation of the cost of a design. Guidelines, both procedural and providers of design assistance are also included within the IDCT Tool domain.

We believe the transformations from design variables to cost metrics do not typically occur in a single step. Therefore, we have further defined the evaluation function’s transformation process, as shown in the shaded box in Figure 4-1. Cost metrics are the result of a series of intermediate costing “elements,” (e.g. design variables, cost measures, cost drivers, product and process features and attributes) and transformations. For example, customer requirements are in some way, possibly using Quality Function Deployment, transformed to design variables. Ideally, customer requirements would be explicitly linked to performance measures, in this case, cost of the design of the product. However, it is very important that cost measures ultimately be linked to variables that are under the control of the designer. The possible transformations are represented in Figure 4-10 by small shaded boxes that contain an "X." The sequence of transformations can also be represented as a series of mappings:

$$r_i \Rightarrow v_i \Rightarrow a_i \Rightarrow d_i \Rightarrow m_i$$

$$R = \{r_1, r_2, ..., r_j\};\ r_i \subset R$$

$$V = \{v_1, v_2, ..., v_k\};\ v_i \subset V$$

$$A = \{a_1, a_2, ..., a_l\};\ a_i \subset A$$

$$D = \{d_1, d_2, ..., d_n\};\ d_i \subset D$$

$$M = \{m_1, m_2, ..., m_p\};\ m_i \subset M$$

Reading the transformations from right to left, the performance measures for design *i*, i.e. a subset of measures m_i from a larger set M, are based on a set of drivers (d_i). The drivers are based on a set of product/process attributes (a_i) which are controlled by a set of design variables (v_i) and customer and company requirements (r_i). This is similar to the flows that occur in the "using" cost evaluation processes described in a previous section.

The costing elements and transformations illustrated in Figure 4-10 raise a variety of research issues. The first issue is *what* are the design-relevant determinants for each element during each phase in the product's life cycle; also, *which* elements of the determinants are relevant *when* in the design process. The second issue is *how* are the determinants of the elements *related* (transformation function); also *which* relationships are applied *when* in the design process. The third issue is *what* are the appropriate methodologies to model and manage the cost evaluation elements and their relationships. The final issue is *how* are the methodologies integrated into the design process. This includes user interface and report design, interoperability among disparate systems, as well as structuring, capturing, representing, processing, and managing the extensive set of knowledge and data bases that are utilized by the system. An effective IDCT Tool should provide a platform and means to address these issues.

The real value from the development and use of an IDCT Tool would be the insight into how design and programmatic variables influence cost. This would ultimately lead to, borrowing from quality management, *designing in* rather than just "inspecting" or checking affordability. The processes of developing and using an IDCT Tool should identify cost and other data needs that are required to effectively support affordability design decisions. These processes should also result in specifications for relevant and timely information that is specifically *for* engineering, rather than a by-product of external financial reporting, operations management, etc. These needs should then lead to a "re-engineering" of costing approaches and data requirements. The pressure for such information should be similar to that exerted on cost accountants by operations managers, which resulted in such new approaches as Activity-Based Costing, Throughput Accounting, etc. Development of the IDCT Tool provides an opportunity to freshly examine how costing *should* be done in order to support design and how new costing methodologies themselves must be *designed* to fit into tomorrow's environment.

4.5.11.2 Learning function

We have included the learning function in the conceptual foundation in order to provide the most complete definition of an IDCT Tool. However, at this time, the learning function is not well defined. This is an area for significant further research. In current practice, the evaluation function is by far the most understood. While learning does take place and evaluation methodologies are developed and modified based on new experiences and information, this process is much more ad hoc. Subsequent projects should identify and define the learning processes and provide tools for effectively carrying out this function.

4.5.12 Object-Based Approach to IDCT Development

Figure 4-11 illustrates the preliminary object-based approach for the IDCT Tool. In this illustration, the focus is on assessing the manufacturing cost of a design; however, the proposed tool would address life-cycle cost. In the approach shown in Figure 4-11, the design is specified through the use of five hierarchies; object-based hierarchies are used to organize and structure the design variables and parameters. Form, material, and manufacturing process variables are defined through the three object-based hierarchies that are illustrated in the left portion of Figure 4-11. Design variable values are property/attribute values of an instance of an object contained either in a generic hierarchy (e.g. material density, cost per pound) or an instance of a collection of objects (e.g. manufacturing process scrap rate may be a property of a collection of a material object and process object). The design is also specified through an item hierarchy or structure, as illustrated in the top portion of Figure 4-11. The item hierarchy considers the design in a larger context; it captures the relationship of the item being designed to the total product (analogous to a work breakdown structure or bill of materials). Through the use of a hierarchical structure, product-related values are inherited by the individual items, e.g. total number of units of the product to be produced, year-dollars for cost estimates. The final hierarchy that is used to define the design is the cost structure, illustrated in the top-right portion of Figure 4-11. The cost structure defines the types of costs that are considered in the design, e.g., life-cycle, manufacturing, recurring. The cost structure also provides the means to “roll-up” and “cut” costs in a variety of ways, e.g. total product cost, investment cost only.

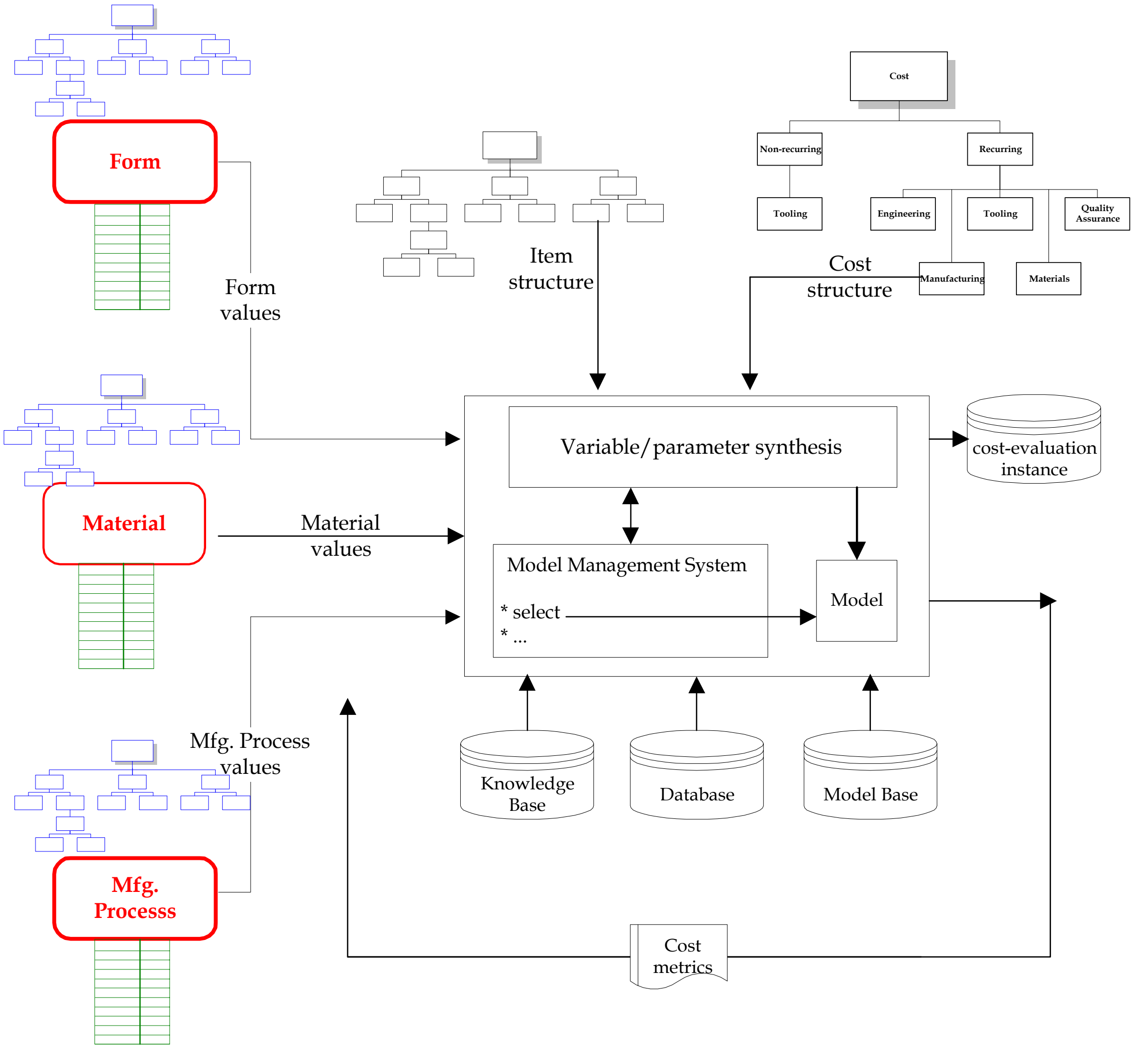


Figure 4-11: The IDCT Tool’s object-based approach to cost evaluation.

The large box in the center of Figure 4-11 illustrates some of the actions that would occur in a typical application of the IDCT Tool. For example, when a design’s cost is to be evaluated and if no model has been specified, the system would select the most appropriate model based on the metric desired and the information available. If there is insufficient information to utilize any model, the Tool would determine what information is needed and request that information of the designer/IPT. This process makes extensive use of the Tool’s access to data, model, and knowledge bases. Also note in Figure 4-11, each evaluation is both reported to the user and recorded for future use and learning (a cost-evaluation instance).

Figure 4-12 illustrates the concept of a cost-evaluation instance. In this case, the instance has two primary database links, one to the item hierarchy and one to the cost structure/hierarchy. The instance is actually linked to the lowest level in the item hierarchy; in the case of Figure 4-12, it is linked to a component. Most of the component’s characteristics are defined through three hierarchies – form, material, and manufacturing process. The component also inherits all of the characteristics from the subsystem and system levels; all of which is available to any linked cost model. A cost model is linked through the cost structure. Each cost instance corresponds to a cost element (e.g. direct labor, materials, overhead) in the cost structure. Similarly, each model is linked to the set of cost elements for which it provides estimates.

One of the characteristics of the models is the inputs that are required for estimating. Therefore, each cost instance links a cost model estimate to the specific input values that generated the estimate.

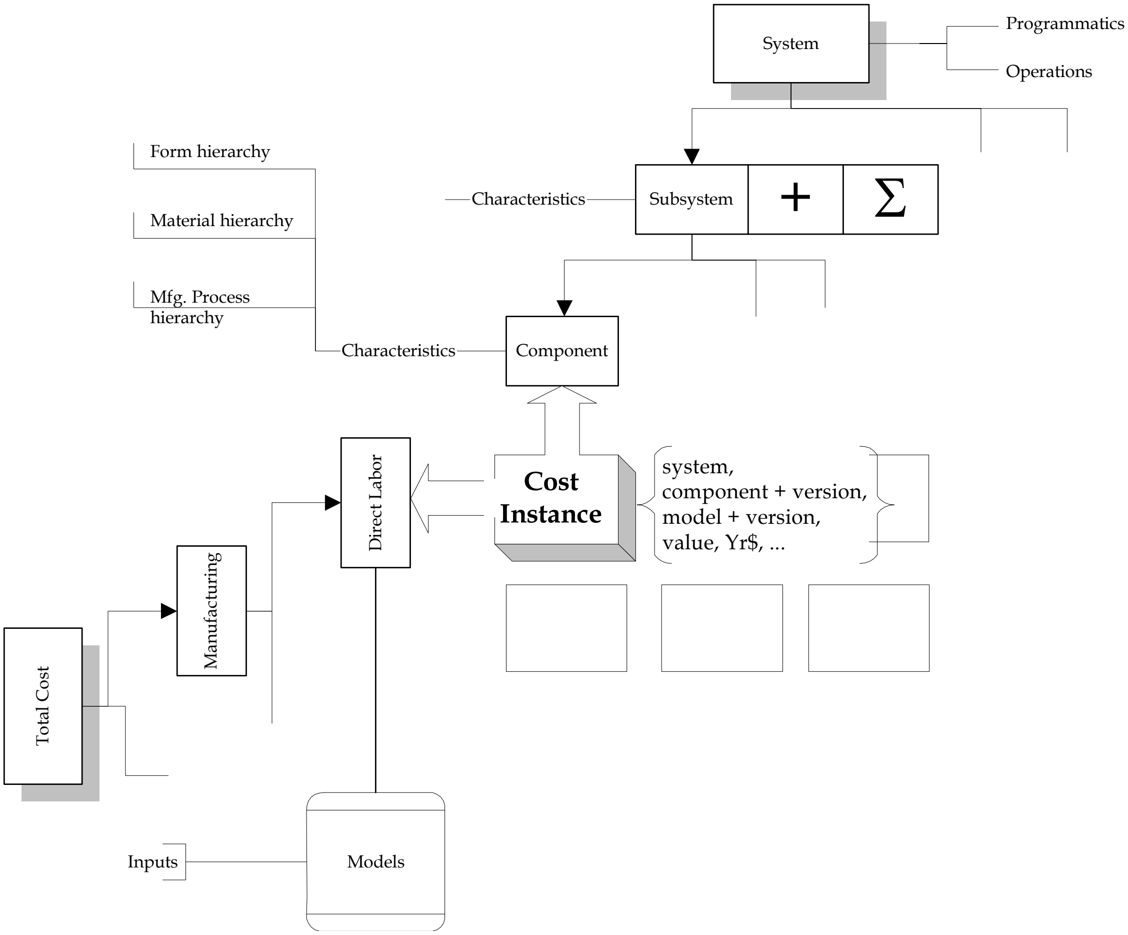


Figure 4-12: Definition of a cost instance, a primary data element in the IDCT Tool.

The above structure supports many of the needs of an effective IDCT Tool, such as documenting the evolution of a design and tracking changes over time due to changes in the design variables, value of the design variables, models, requirements, data sources, etc. Models may be parametric, detailed build up, or actual production/operations data. The cost instance concept also facilitates searching for and identifying comparable items that have been designed in the past.

The concepts described above are built upon the use of object-based technologies in order to structure most of the information that is required to evaluate the product/process cost of a design. As shown in both Figures 11 and 12, the design variables, a primary set of inputs to the cost-evaluation process, are divided into three categories that represent the *product triad* – form, material, and process; i.e., evaluation of a design's cost must concurrently consider the item's form, material, and process (how it will be produced). This concept needs to be expanded to include a "use" category in order to be able to assess life-cycle cost.

The flexibility inherent in the object-based approach allows different users to represent, manage, and query data, objects, methods, output, etc. that best meets their needs. That is, a system can easily be redefined or reconfigured to best represent a specific product, project or company organization, level of expertise, design requirements and considerations, etc. Similarly, the object-based structure accommodates increasing levels of detail that result as a design evolves.

Examples of the types of structures and hierarchies that are used in the IDCT Tool are shown in Figure 4-13. These include a portion of a manufacturing processes hierarchy in the left-hand portion of Figure 4-13 and a portion of a cost structure in the right-hand portion of the figure.

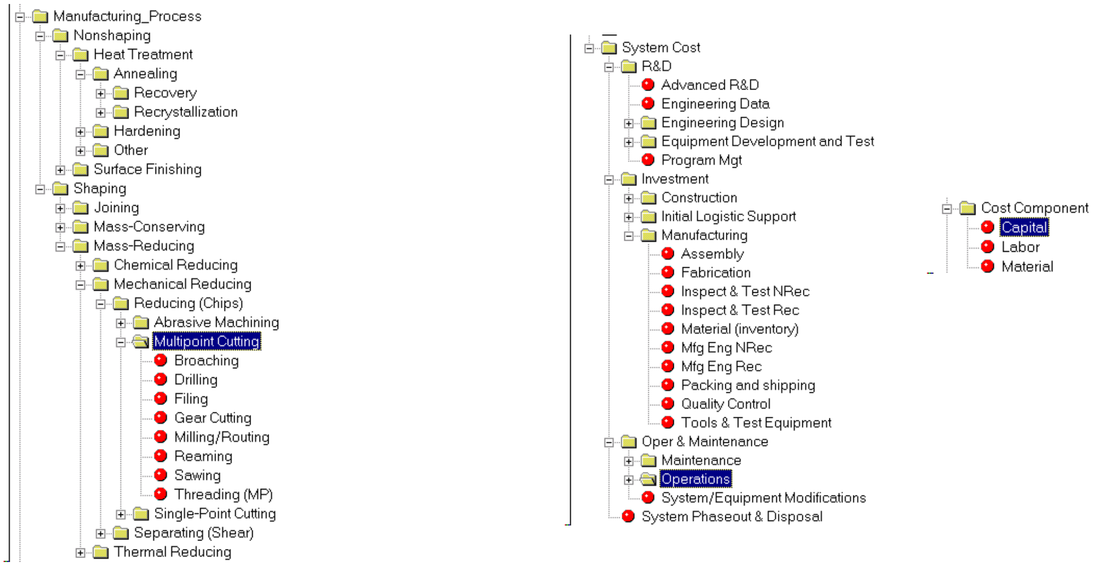


Figure 4-13: Example hierarchical structures used in the IDCT.

4.5.13 Concept of Operation of an IDCT Tool

The concept of operation (ConOpt) of an IDCT Tool is a description of how it will be used. While there are many potential application of the IDCT Tool, we focus on a primary application -- determining the cost of a design alternative that is under consideration. The ConOp provides the basis for the design and development of the system; the system development process is much more effective when it is based on a clear, well-defined ConOp. Since the objective of the IDCT Tool is to facilitate and enhance the design trade study processes, the initial definition of the ConOp is the general trade study process that was defined in Figure 4-5. A more detailed view of Figure 4-5 is provided below in Figure 4-14. Figure 4-14 decomposes the “Analyze Candidate Solution” activity from Figure 4-5 into its sub-activities: determine performance, determine producibility, determine supportability, and determine cost of the design being considered.

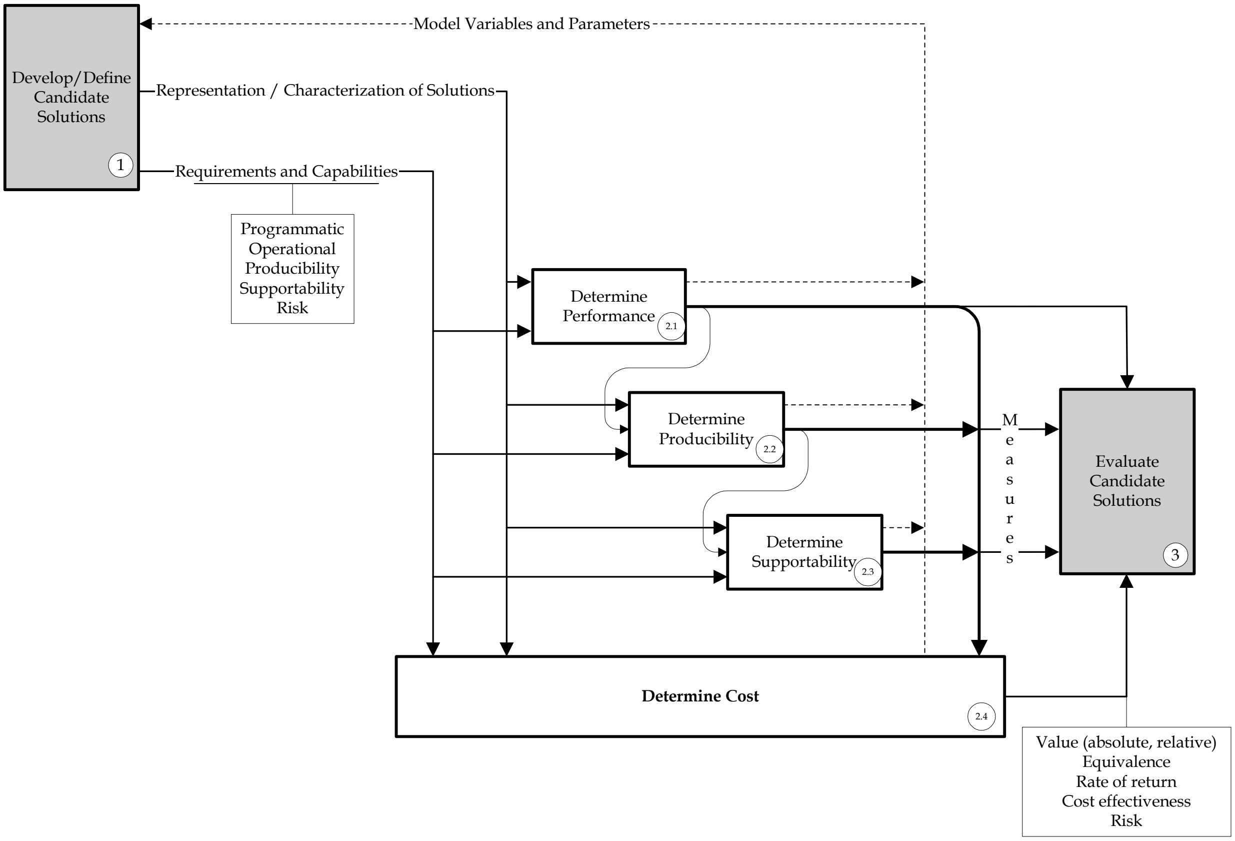


Figure 4-14: Decomposition of "Analyze Candidate Solution" activity.

In order to understand how the IDCT Tool will provide cost evaluations of candidate designs, the "Determine Cost" activity (numbered 2.4) in Figure 4-14 is decomposed into its sub-activities; that decomposition is shown in Figure 4-15. The "determine cost activity" is represented in a cross-functional process view in Figure 4-15; i.e., each sub-activity is associated with one or more organizational units. Each organizational unit is represented as the horizontal "swim lane" in Figure 4-15. The primary sub-activity, at least for the cost engineer, is "utilize cost model" (labeled 2.4.1).

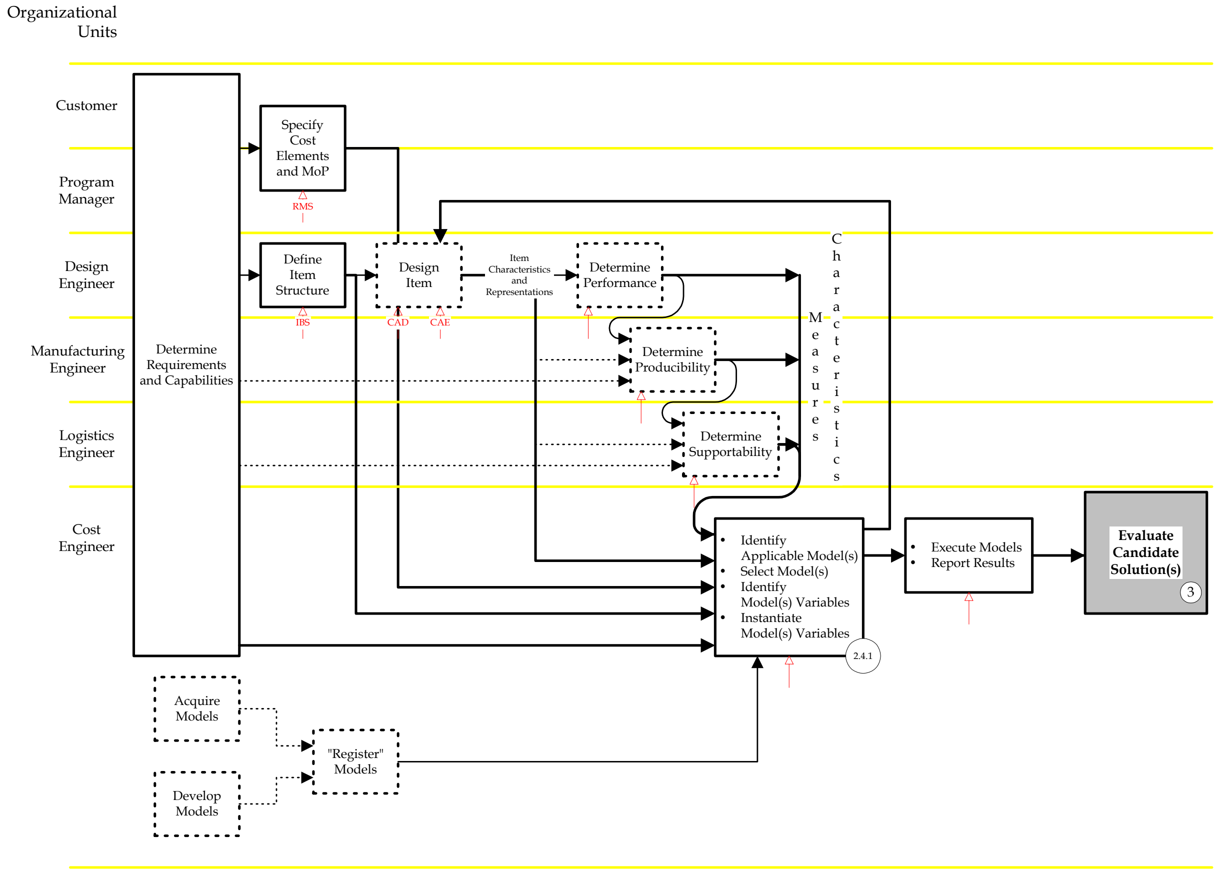


Figure 4-15: Decomposition of the "Determine Cost" activity.

The "utilize cost model" activities are shown in Figure 4-16. In order to utilize a cost model, the cost engineer (or an automated system) must identify applicable models based on the information available (obtained from item structure, cost structure, requirements and capabilities, representations/characterization of solutions, and other measures) as well as the characteristics of the model, as shown in the left-hand portion of Figure 4-16. A model is applicable if it provides the type of cost desired, is based on variables that currently have values defined, etc. The model must also be "registered" with the system; i.e., the system knows what costs it provides, what variable information it requires, etc. However, the system does not need to know how the model works; i.e., it only needs to know what the model does, not how it does it. Once a set of candidate models are identified, one or more are selected for use based on specified criteria (e.g., how well the model has performed in the past, how closely it matches the data available). The variables and parameters that are used by the model are then identified and subsequently instantiated -- values for the model's variables are provided by the user or though automatic links to databases, CAD system, etc. The model is then executed and results reported.

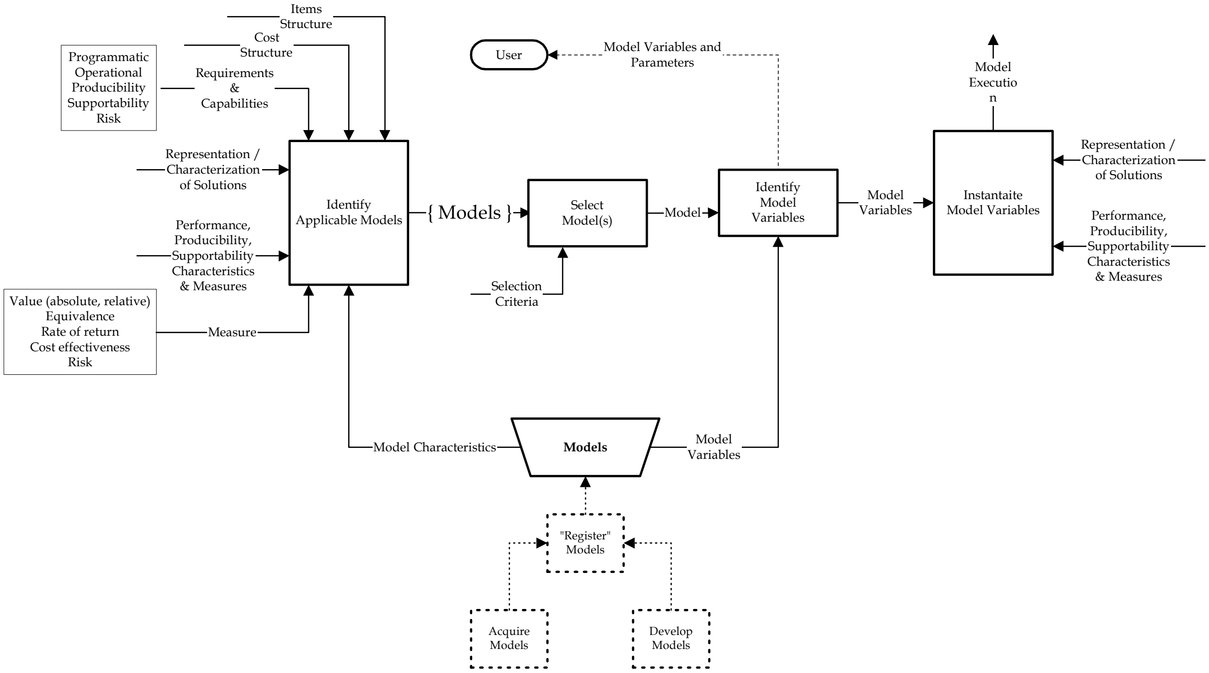

Items Structure
Cost Structure
Programmatic
Operational
Producibility
Supportability
Risk
Requirements & Capabilities
Representation / Characterization of Solutions
Performance, Producibility, Supportability Characteristics & Measures
Value (absolute, relative)
Equivalence
Rate of return
Cost effectiveness
Risk
Measure
Identify Applicable Models
{ Models }
Select Model(s)
Selection Criteria
Model
User
Model Variables and Parameters
Identify Model Variables
Model Variables
Instantaite Model Variables
Model Executio n
Representation / Characterization of Solutions
Performance, Producibility, Supportability Characteristics & Measures
Model Characteristics
Models
Model Variables
"Register" Models
Acquire Models
Develop Models

Figure 4-16: Decomposition of “Utilize Cost Model” activity.

This overall process is summarized in Figure 4-17. The two shaded cloud-like symbols in Figure 4-17 represent the design/IPPD decision-making and trade study process and the cost-evaluation process. Specific activities from the design/IPPD process that are directly related to cost evaluation are identified by the process flow diagram under the design/IPPD “cloud.” Similarly, the cost-evaluation activities that are directly related to the design trade study process are identified by the process flow diagram under the cost-evaluation “cloud.” The IDCT Tool provides the link between the two sets of processes. The heavy vertical arrows show the primary links and flows of information between the two sets of processes.

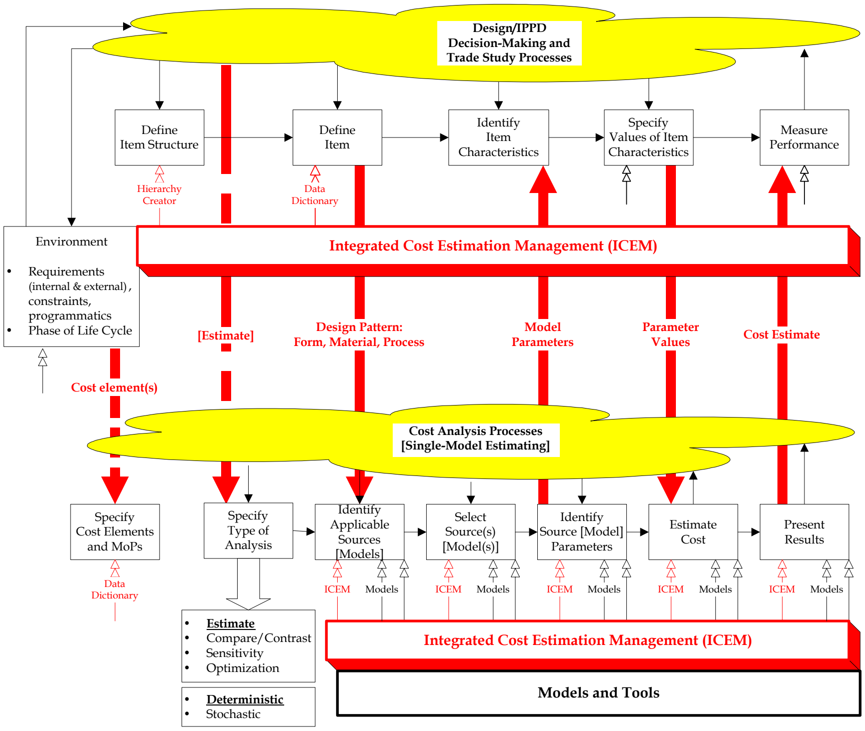


Figure 4-17: IDCT Tool links design/IPPD and cost-evaluation processes.

A portion of the activities from Figure 4-17 is isolated in Figure 4-18. This shows a flow similar to the one in Figure 4-16 that illustrated the "utilize cost model" activity. It is essential that all of the processes that the IDCT Tool will support be defined in a manner similar to those shown here.

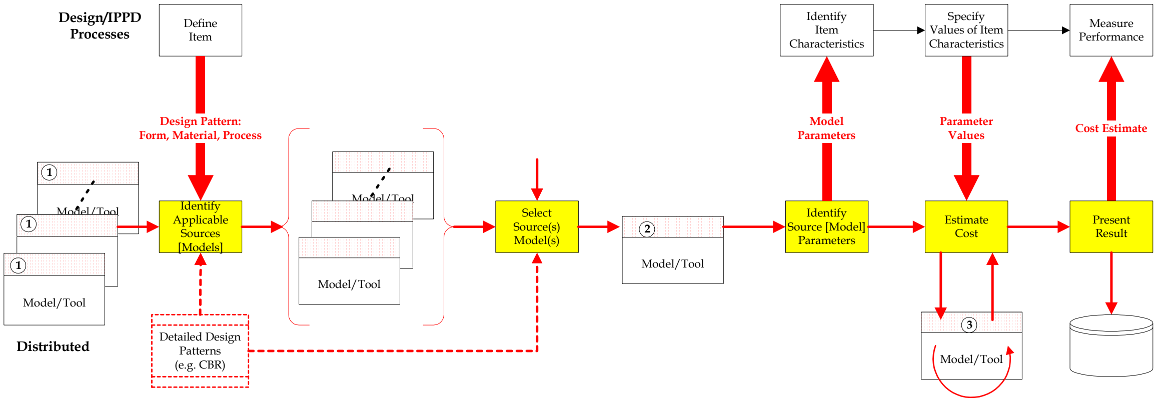


Figure 4-18: A portion of the cost-evaluation process and its links to the design process.

So far all of the illustrations have dealt with recurring activities, those that happen each time an item's cost is evaluated. Figure 4-19 provides an illustration of a non-recurring activity – specifying the cost elements in order to initiate an cost evaluation. As shown in Figure 4-19, the user specifies the cost elements and measure(s) of performance based the project requirements, desired cost type (selected from a standard cost structure) and the desired cost component(s) (e.g. labor, material, capital, etc.).

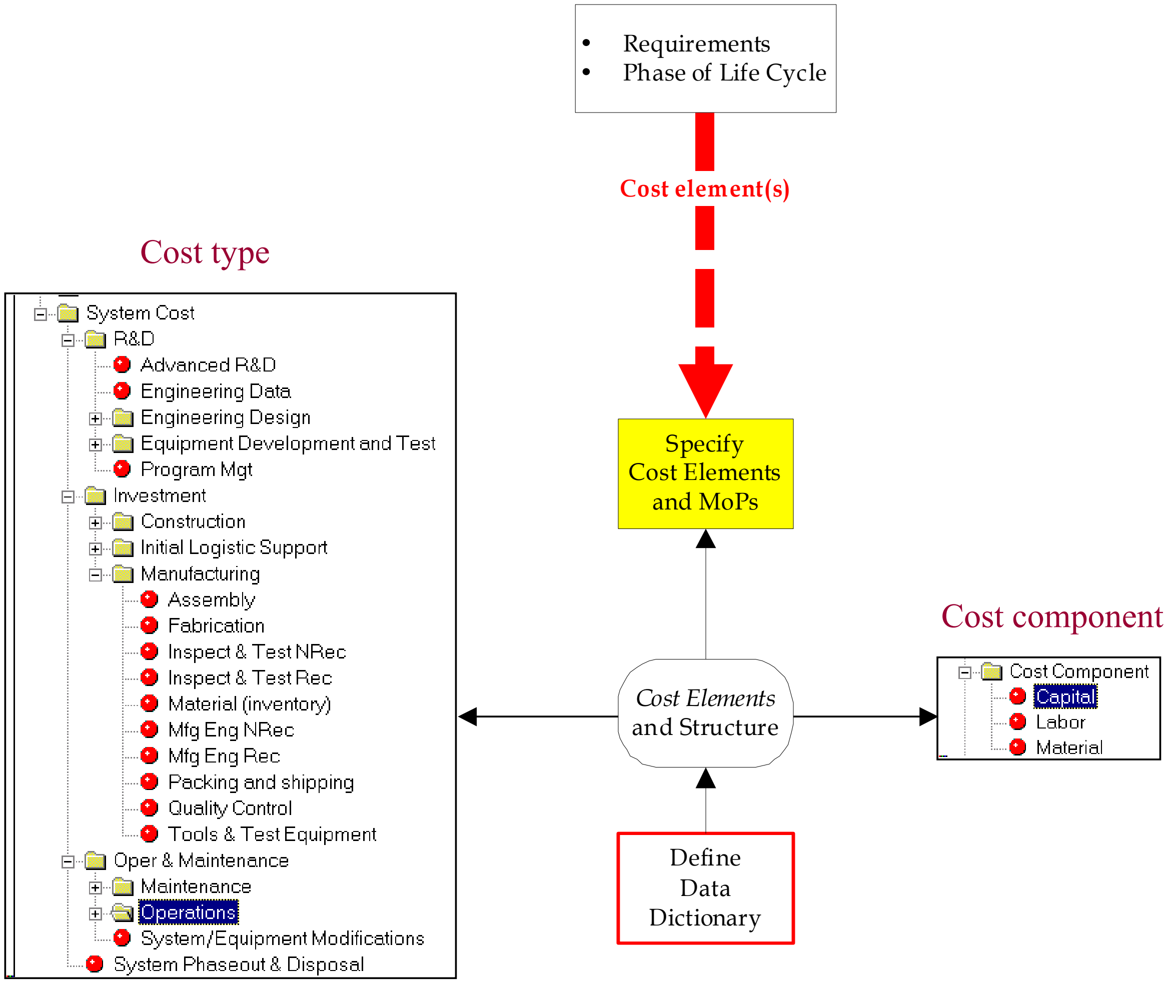


Figure 4-19: A portion of the cost-evaluation process and its links to the design process.

4.5.14 Intelligent Integration for Cost Estimation

In order to effectively perform the evaluation function (and the learning function), the focus of the development of the IDCT Tool must be on integration and intelligence.

4.5.14.1. Basic Concepts of Intelligent Integration

Integration is required in order to link the islands of cost estimation tools and relevant information systems and provide a variety of cost estimation services. Integration should provide the designer/Integrated Product Team (IPT) with a choice of cost estimation tools, access to the most current models and data, and the capability of utilizing different cost estimation tools within a design in order to supplement and complement each other.

Intelligence is required to support both integration and the provision of cost estimation services. In terms of integration, the IDCT Tool should enable interoperability among different cost estimation tools and associated information systems, e.g., CAD systems, product data management (PDM) systems, and activity-based costing (ABC) systems. The IDCT Tool should provide commonly accepted semantics (i.e., the representation of product, design, process and the other cost-estimation related information) and open communications.

Intelligence is also required to support such cost estimation services as model discovery and retrieval and cost estimation by analogy. Since the IDCT Tool will have access to many cost estimate tools/models, model discovery will facilitate the identification and selection of applicable models based on the type and amount of design information available. Estimation by analogy provides cost estimates for a design that are based on entities with similar design characteristics and in which cost data are available.

Figure 4-20 provides a conceptual view of an Intelligent Integration Scheme (IIS) for the IDCT Tool. There are two basic components in the Intelligent Integration Scheme: Integrated Cost Estimation Management (ICEM) and Integration Framework.

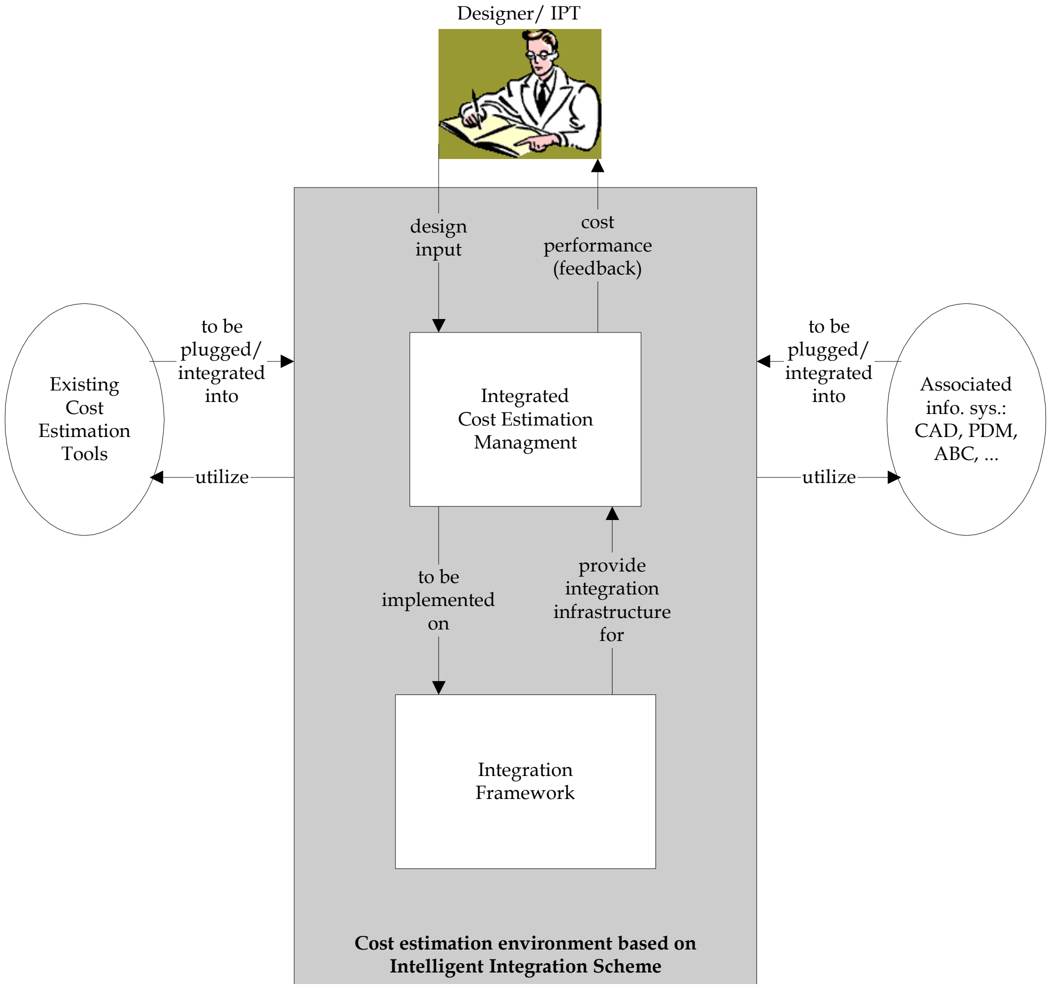

Designer/ IPT
design input
cost performance (feedback)
to be plugged/ integrated into
utilize
Existing Cost Estimation Tools
to be plugged/ integrated into
utilize
Associated info. sys.: CAD, PDM, ABC, ...
Integrated Cost Estimation Managment
to be implemented on
provide integration infrastructure for
Integration Framework
Cost estimation environment based on Intelligent Integration Scheme

Figure 4-20: Conceptual view of an Intelligent Integration Scheme.

Integrated Cost Estimation Management (ICEM) is the primary component of the Intelligent Integration Scheme. It provides direct services for designers and IPTs, including basic cost estimation, hierarchical design/cost representation, cost aggregating, cost/performance trade-off management, design guidance, intelligent cost model discovery, cost estimation by analogy, computer supported collaboration work (CSCW) services, etc. Any cost estimation tools that is integrated into the framework would have access to these services, i.e., they would "reuse" portions of the ICEM. Therefore, it would not be necessary for the other cost estimation tools to develop and implement advanced cost estimation services that are included and available in the ICEM.

Development of the ICEM requires: (1) definition of cost estimation processes (i.e., the way that cost evaluation is carried out during design), (2) identification, definition, and development of user-oriented cost-estimation/design services, and (3) specification of the types of interfaces required by the various user groups. The ICEM would be implemented on top of the Integration Framework.

The Integration Framework provides the foundation for implementing the ICEM. It provides the interconnections and interfaces among cost estimation tools and relevant application systems. The Integration Framework provides commonly accepted data representations, data dictionaries, and interfaces between the different cost estimation tools and associated information system; i.e., it allows the required technologies to be "plugged into" a common costing environment with minimal effort.

Development of the Integration Framework requires: (1) identification of middleware, network protocols, and communication architecture for effective integration of different cost estimation tools and associated information system (This is necessary to enable distributed and heterogeneous tools and systems to invoke one another's resources.), (2) identification, definition and development of standard data representations and a data dictionary for all cost-estimation-related information that is to be shared (This will result in commonly accepted semantics within the integration environment and enable distributed and heterogeneous tools and systems to understand each other's data.), and (3) definition of interfaces between components, i.e. the primitive operations and services that are required by the user-oriented cost estimation services. For example, services that are useful for managing (e.g., retrieving comparable designs that are the most "similar"), coordinating (e.g., grading comparable designs in terms of similarities, logging which tools have been used under what circumstances and with what result), and controlling (e.g., obtaining the properties of a design, invoking an estimation request, obtaining CAD drawings or attributes) cost estimation activities.

Object-oriented methodologies provide the means for developing the scheme. Different cost estimation tools and relevant information applications and the ICEM are considered separate modules (also called components, objects) inside a costing environment. Object-orientation provides such advantages as: reuse of software components, faster development, reduced development risks for complex systems, easier maintenance, and increased quality etc. (Hathaway, 1996).

4.5.14.2 Critical Enabling Technologies

In order to realize the transparent integration of different and disperse cost estimation systems as discussed above and to enable distributed and heterogeneous applications to invoke one another's resources and to understand each other's data, appropriate technologies must be identified and utilized. Two basic requirements to meet these needs are code interoperability and common semantics. Standards (formal and/or de facto) play an important role in developing these technologies.

The Object Management Group (OMG, 1999) was founded in 1989 in order to develop a set of standards that would facilitate distributed computing and guarantee application and code interoperability. As a result, the Common Object Request Broker Architecture (CORBA) has emerged as a de facto standard for interoperability among the rapidly proliferating hardware and software products that are evolving today. OMG has defined the Reference Model Architecture (RMA), upon which distributed applications can be constructed. The core of this architecture is the Object Request Broker (ORB), which is a common communication bus for objects. Through an ORB, a client can transparently invoke a method on a server object, which can be on the same machine or across a network. The ORB intercepts the call and is responsible for finding an object that can implement the request, pass it the appropriate parameters, invoke its method, and return the results. The client does not have to be aware of where the object is located, its programming language, its operating system, or any other system aspects that are not part of an object's interface. In so doing, the ORB provides interoperability between applications on different machines in heterogeneous distributed environments and seamlessly interconnects multiple object systems.

ORBs simplify the process of getting two applications, new and existing, to inter-operate with each other. This is accomplished through a language-independent (neutral) interface specification (CORBA, 1998) – the Interface Definition Language (IDL). The IDL allows programmers to choose the most appropriate operating system, execution environment and even programming language to use for each component of a system that is being developed. With the support of CORBA, a legacy cost estimation system can be "plugged into" a proposed environment through modeling its interface with the IDL and then writing the corresponding "wrapper" code.

In addition to code interoperability, an effective integration requires the mutual understanding of product, design, process and the other cost-estimation related information. STEP (ISO 10303, The Standard for the Exchange of Product Data, 1997) is and open and extensible international data description standard, which will provide a complete unambiguous, computer-interpretable definition of the physical and functional characteristics of a product throughout its life cycle. For a large set of manufacturing products and processes, STEP definitions are available; others are under development and are expected to be available soon.

Therefore, both CORBA and STEP are technologies that are required by an Intelligent Integration Scheme. It is anticipated that they would be implemented in two different layers in order to guarantee the open feature of the integration framework. The CORBA layer would provide support for the heterogeneous and distributed applications to use one another's resources through code interoperability and the STEP layer would allow the semantics of manufacturing information to be understood by multiple applications.

As described above, intelligent cost model discovery and intelligent cost estimation are critical services required by the Intelligent Integration Scheme. They involve the selection of the appropriate model(s) and data based on the design information that is available. The selection process is quite similar to one type of human problem-solving logic — base a solution on one that has worked for a similar problem in the past. Case-based reasoning (CBR), also called analogy-based reasoning, is an artificial intelligence technique that can be used to realize this type of reasoning.

CBR's approach to problem solving is based on the retrieval and adaptation of cases, or episodic description of problems and their associated solutions (Allen, 1994). In CBR, a problem is solved by searching previously encountered cases, retrieving similar cases and approaches and modifying them if necessary to fit the current problem. CBR has been a very effective tool in practice for developing knowledge-based system. CBR is easier to implement than rule or model-based approaches because the cases represent concrete examples which are easier for users to understand and apply than complex chains of reasoning generated by rules or models. Learning in CBR occurs as a natural by-product of problem solving. When a problem is successfully solved, the experience is retained in order to solve similar problems in the future. When an attempt to solve a problem fails, the reason for the failure is identified and remembered in order to avoid the same mistake in the future. In cost estimation, each case corresponds to a design alternative and the experience is the cost performance of the design. The process involved in CBR can be represented by a schematic cycle comprising the four “re”s: (1) retrieve the most similar cases, (2) reuse the case(s) to attempt to solve the problem, (3) revise the proposed solution if necessary, and (4) retain the new solution as a part of a new case.

In terms of cost estimation, CBR can be applied follows. When a new design requests a cost estimation service, the most similar designs from all previous designs are retrieved. In the reuse phase, the cost model of the retrieved designs is adapted to the new design. The cost of the new design is estimated with the adapted model. Through the revise process, the estimation result is tested for success, e.g. through future feedback from actual production or support data, and is “repaired” if it “fails.” Successful cost models are then retained for future reuse, and the case base (cost models base) is updated by the new model or by modification of some existing cases.

4.5.15 Preliminary Design of an Integration Framework

A preliminary design of the integration framework is shown in Figure 4-21. It uses a typical web-based three-tier client/server architecture. The World Wide Web (WWW), a hypertext based document system that acts as a distributed information service, encapsulates communications protocols to organize and access data across the Internet. Due to the Internet’s widespread acceptance and penetration, the communications infrastructure based on the WWW provides the most cost-effective and portable solution for a cost estimation environment.

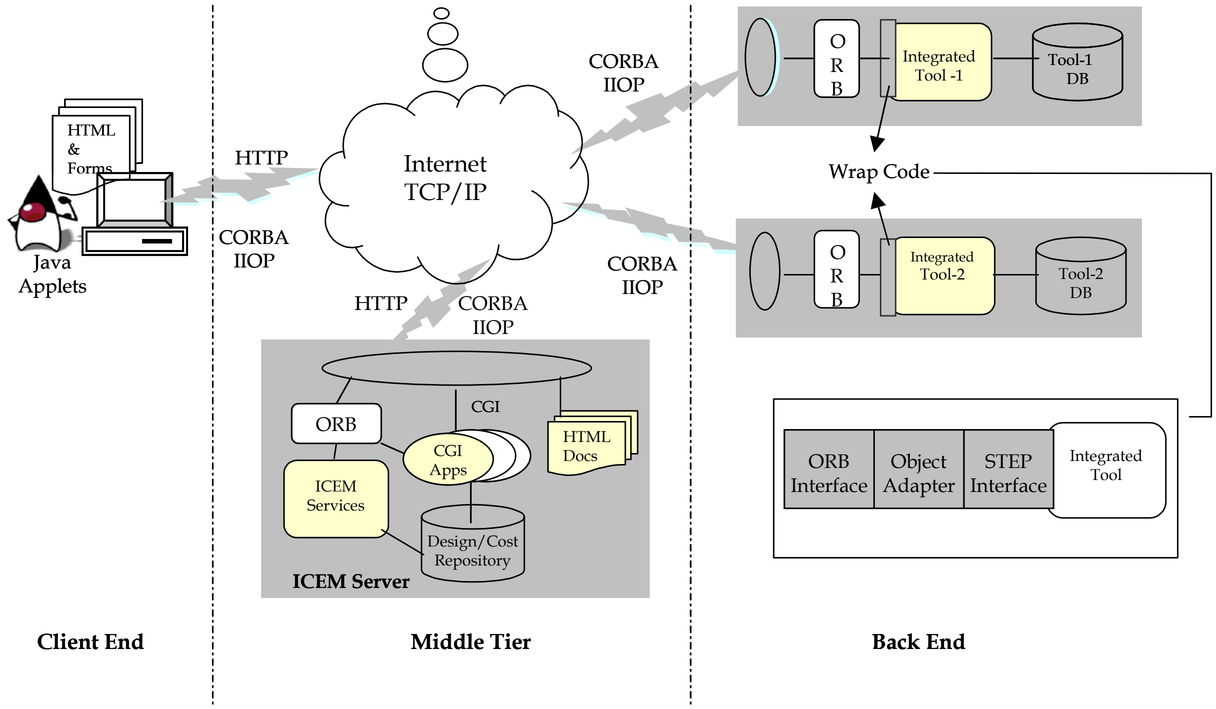


Figure 4-21: Preliminary Design of an Integration Framework for an IDCT Tool.

The overall system is component based. Client Java components interact with other Java components as well as with server components. Conversely, server components invoke methods on client components using CORBA events and callbacks. HTTP (HyperText Transport Protocol), the communication protocol for moving hypertext files across the Internet, is used to establish a connection with a Web server and transmit HTML pages to a client browser, download Web pages, Java applets, and images. CORBA and IIOP are used for client-to-server and server-to-client communications. CORBA provides a way to execute programs written in any language regardless of where they reside on the network or what on platform they run. IIOP (Internet Inter-ORB Protocol) is the CORBA message protocol that is used on a TCP/IP network. IIOP links CORBA's General Inter-ORB protocol (GIOP) to TCP/IP and specifies how CORBA's Object Request Brokers (ORBs) communicate with each other. TCP/IP (Transmission Control Protocol/Internet Protocol) is a set of network communications protocols that is used by computers on the Internet.

In Figure 4-21, the Client End (the first tier) belongs to Web browsers with Java support. It include a mixture of Java and HTML applications, two cross-platform languages that provide the convenience of “write once, and run everywhere.” The client end provides the interface between the user, e.g. designer or IPT, and cost-estimation objects. This is done via Java applets (small applications written with Java and running at the web browser) and HTML forms; i.e., CGI (Common Gateway Interface) applications. CGI is a standard for interfacing or providing a gateway to an external application (such as a database server) with a WWW server (a machine that runs an HTTP daemon). It can provide information to, and accept information from, WWW browsers (such as Netscape) anywhere on the Internet.

The Middle Tier (the second layer) runs on any server that can service both HTTP and CORBA clients. The CORBA objects on the server interact with each other using a CORBA ORB. The ICEM (Integrated Cost Estimation Management) component resides in this tier, as well as a design/cost repository that is required by the ICEM to store interfaces with registered cost estimation tools and related applications, historical design/cost information, and such environment information as user information, registered services etc.

The Back End (the third tier) is where the actual encapsulated cost estimation components reside. As shown in Figure 4-21, all joined cost estimation tools will be built and packaged as components, i.e., using a CORBA Interface Definition Language (IDL) to wrap existing code with object interfaces.

Since the overall system architecture conforms to CORBA, it provides a way for heterogeneous, distributed systems to inter-operate with each other, and to integrate them together with minimal effort. More importantly, this integration process is realized transparently between heterogeneous and distributed systems. That is why Object Request Broker (ORB) serves as the major communication bus inside this integration framework. All cost estimation tools and relevant application systems can be integrated into this framework simply through using a CORBA IDL to wrap existing code from any language. Object adapter and the ORB interface results from such wrapping. With the support of CORBA, the integration framework can easily be extended to become a more comprehensive integrated platform for Integrated Product/Process Development (IPPD).

As discussed above, the Standard for the Exchange of Product Data (STEP) provides an integral and widely accepted neutral language to describe much of the information required for cost estimation. Therefore, another wrapping layer for the STEP interface is required for each cost estimation tools and associated applications.

The CORBA-based integration scheme is illustrated in Figure 4-22. The communication between the IDCT Tool, shown in the client host, and a cost-evaluation model, shown as an object on the server host, is through an ORB (object request broker) using the Internet InterORB Protocol (IIOP). The IDCT Tool's link to the model is through the model/tool interface and wrapping code. The IDCT Tool only needs to know the type of cost-estimation service that the model/tool provides and the definitional objects it uses to provide that service; it does not need to know how the service is provided or the details of the model/tool implementation.

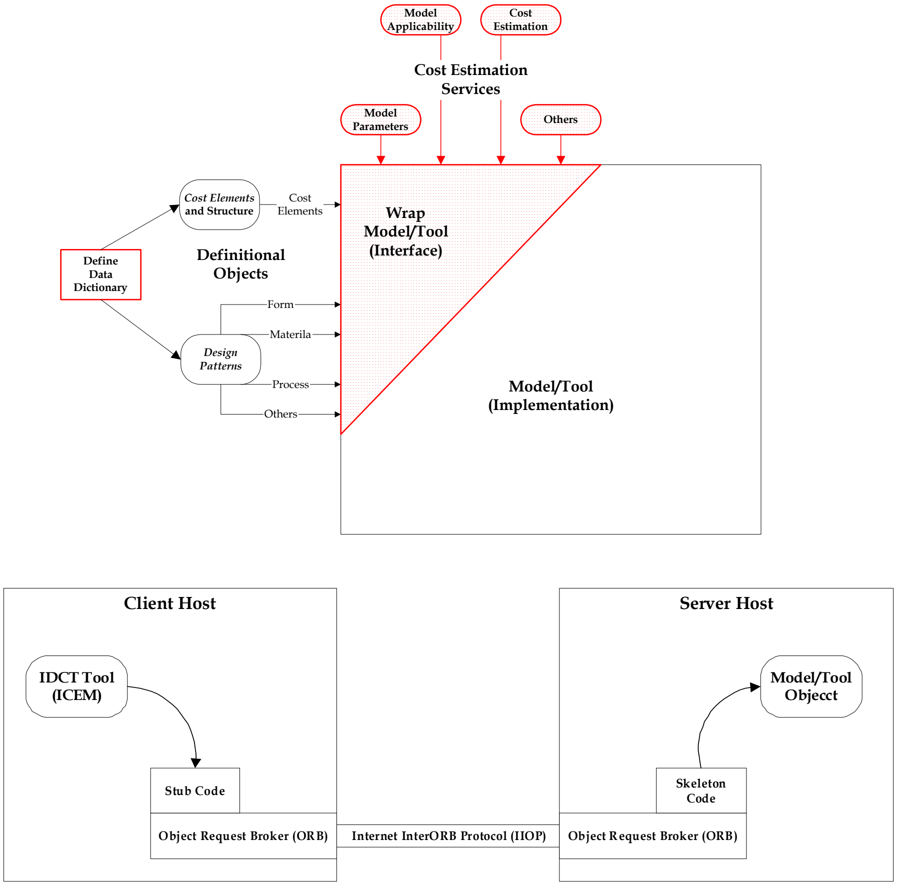


Figure 4-22: The IDCT Tool's CORBA implementation scheme.

4.5.16 Prototype of an IDCT Tool

The prototype is a CORBA (Common Object Request Broker Architecture) based application. The development depends strongly on the ORB (Object Request Broker) product. Currently there are two major products in this market: IONA's Orbix and Inprise's (bought by Borland) Visibroker. We adopted Orbix at first for its large market share and customers such as Boeing. In order to develop the prototype, we obtained an evaluation version of Orbix 3.0 (C++ ORB binder) and OrbixWeb 3.2 (Java ORB binder).

The development activities of the prototype are summarized in four phases:

- We built a small CORBA application using a free Java ORB from Sun Micro systems in order to gain proficiency in CORBA development and to evaluate the inter-operability of CORBA. This phase was successful. We developed a small database object on the server side and a simple query interface on client side. The client-side code that resides on an NT system successfully activated server objects that reside on a Sun Solaris system.

- We scripted the required objects for the ICEM system using IDL and developed an overall ICEM implementation scheme. The IDL is provided as Appendix 3.

Objects such as Cost, Item, Assembly, etc. are common objects that would be used extensively in the ICEM services. The intent of our design is to implement these objects on the Client Side. Objects such as ModelAgent are typically server-side objects since they commute services to the client. The common objects, (e.g. Cost, Item) are passed back and forth between the server and client. For example, ModelAgent.CostEvaluation(Item) passes the Item object to ModelAgent, which returns the Item object with the refreshed cost object for the specified Item. In this design, the server-side object (e.g., ModelAgent) has to call-back the client- side object (Item) if it requires some functionality from the Item object as it performs its Agent service.

The prototype is targeted to work within the Web environment. Users need to use a web browser to access ICEM services. Therefore, we adopted Java as the client-end language. For the server side, we are using OCEANS as an example cost estimation tool. Since it was built in C++, we have to use a C++ CORBA binder in order to bind it with the ICEM.

- Based on the services we defined for the ICEM, we began to develop the ICEM client with Java Builder. This is the front end that will provide services directly to user. We developed a cross-platform, user-friendly applet front end with the latest Java 2 platform features.

- The client and server side application are bound to the ORB code so that they may co-operate. This involves three separate development activities:
  .1. compile the IDL using the ORB vendor provided language binder to create required Stub/Skeleton code for client/server application,
  .2. bind the client application with the ORB using the stub/skeleton code, and
  .3. bind the server application with the ORB using the stub/skeleton code.

We finished this phase of coding; however, the server side object (ModelAgent) failed to inter-operate with the client-side object (Item), although the invocation process started successfully. The inter-operation is a basic and key capability that we planned to demonstrate; without it, the prototype is basically non functional.

We found that the ModelAgent did inter-operate with client-side object item. It obtained all of the object information; however, it failed when it requested further information that has to be retrieved via the call-back mechanism (the server calls back the client). We sought help from IONA's customer support. Under their request, we sent them a minimal testing case. They are also puzzled with the error and indicated that it may be a bug in their current release. They sent our testing case to Dublin for further analysis; however, so far we have not received a solution from IONA.

As part of the investigation, we identified another possible contribution to the current problem. There are two code binding methods for servants to establish their connection to the corresponding IDL interfaces and thereby to make themselves available to client requests: the inheritance approach, which is recommended in most cases, and the Tie-based approach. The Tie-based approach was the only option for us and we used it extensively in the ORB stub/skeleton code binding. We needed to use the Tie approach because Java has a single-inheritance restriction on classes and the objects that we defined already have inheritance relationships. The disadvantage of Tie approach is that it cannot maintain the defined inheritance relationship; therefore, we have to use hardcoding to maintain the inheritance described by the IDL. This complicated the interface between client and server. It adds complexity in the communication between the client and server in addition to the call-back. For example, using the Tie approach, if we pass a general object (Item or Assembly) to the server side for processing, the pass-in object will not be able to be automatically narrowed to a specific object (Assembly or Unit). As a result, we posed some questions to IONA support in order to clarify our situation. They answered our question in detailed after several exchanges. However they did not give us a solution for the call-back situation. Without help from IONA, it will be extremely difficult for us to analyze and debug the prototype because the skeleton/stub code are created by the IDL compiler which is provided by the ORB vendor. There may be a fundamental problem in using the current implementation of Orbix or with the CORBA.

A lot of time was trying to resolve this issue. In the meantime, we considered another product from IONA, Orbix 2000, which provides full support for POA (portable object adapter), a recommended upgrade for all CORBA applications. We re-compiled the IDL using Orbix 2000 and re-programmed the stub/skeleton code with the POA technology. The same problem resulted.

We have sought advice from other IT sources and have concluded that the problem may be a bug within IONA or an inherent problem in CORBA. The most direct solution at this point is to avoid extensive call-back operations in the CORBA application. This requires a re-design of the overall scheme, requiring least two months, given the current development manpower situation. Another option is a workaround scheme provided by the Chief Architect from IONA. It may provide a quick fix to get our prototype working; therefore, we will try this fix, and if successful, should have a working prototype by November 1, 2000.

The data representations that are used in the IDCT prototype are shown in Figure 4-23.

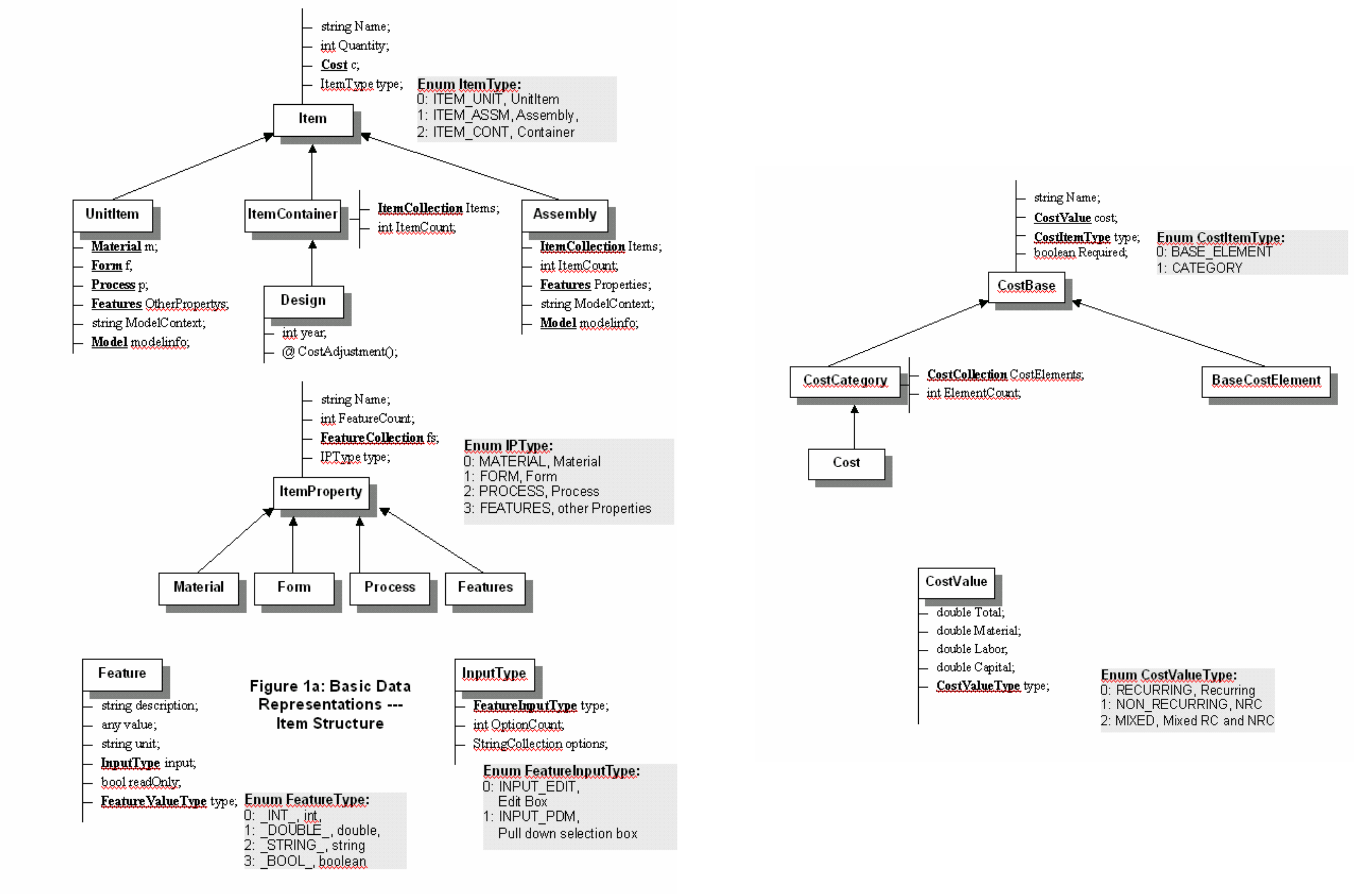
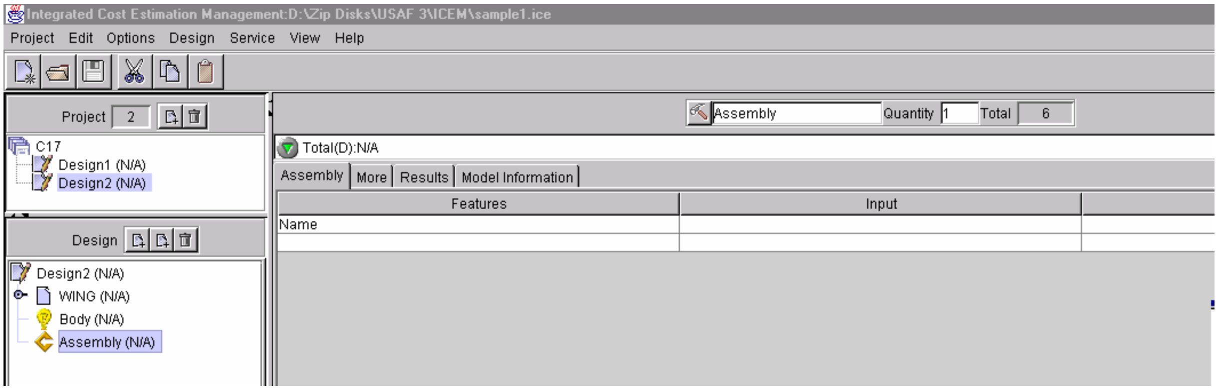


Figure 4-23: Data representations in the ICEM

One screen shot from the prototype is provided in Figure 4-24 provides a glimpse into the user interface of the IDCT; however, this will not be very meaningful until the prototype is functional and the screen shot portrays a more comprehensive populated example. Additional screens shots will be provided as part of the addendum to this report; as discussed earlier, it will be released when the prototype problem is resolved and the prototype is functional.


Integrated Cost Estimation Management:D:\Zip Disks\USAF 3\ICEM\sample1.ice
Project Edit Options Design Service View Help
Project
Design
Assembly
More
Results
Model Information

Project 2

C17
Design1 (N/A)
Design2 (N/A)

Design2 (N/A)
WING (N/A)
Body (N/A)
Assembly (N/A)

Assembly Quantity 1 Total 6

Total(D):N/A

Features	Input
Name	

Figure 4-24: Example screen shot of the IDCT user interface.

4.5.17 Technical Review Board and Project Team

The formulation, design, and development of an In situ Design Cost Trades Tool requires a solid understanding of the cost assessment and design processes, cost-evaluation technologies, and a broad range of supporting technologies. While follow-on project to develop the IDCT Tool would build upon this and other related projects that have been led by the PI, it obviously requires significant involvement from multiple disciplines. In order to meet that need, a diverse team has been assembled that brings a wealth of experience, an impressive depth of knowledge, and a variety of perspectives to bear on the problem.

The team would be led by Dr. Allen Greenwood, Associate Professor of Industrial Engineering, Mississippi State University (MSU), who is the PI for the current project. In addition, Dr. Masoud Rais-Rohani, Associate Professor of Aerospace Engineering, would be an investigator on the project. Several students would also provide research support, including at least two full-time graduate students (one Masters and one Ph.D. student) and at least one undergraduate student.

Table 4-7 provides a summary of the project team members that are not affiliated with MSU. It includes a brief statement of the expertise that they would bring to the project.

The project would be organized as a single team that is composed of four groups, as illustrated in Figure 4-25. Each group represents a key aspect of the project. The Steering Committee, or Technical Review Board, provides overall guidance, direction, and management of the project; it is composed of the PI, customer/sponsor, and senior product/process design and cost professionals. The Design/Cost Process Group provides expertise on general design and cost evaluation methodologies and processes.

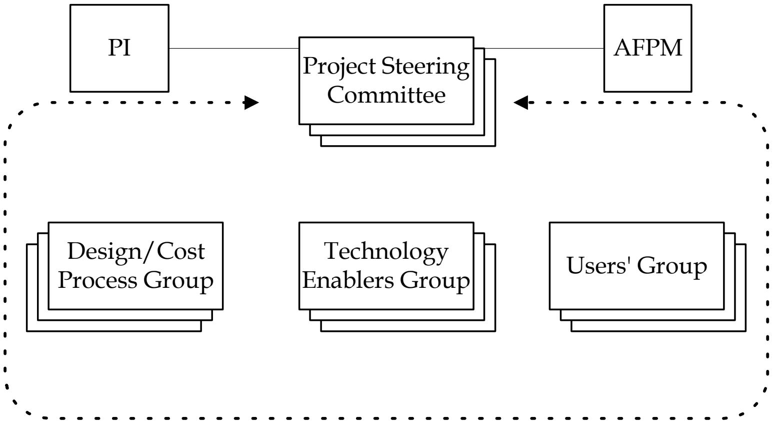

PI
Project Steering Committee
AFPM
Design/Cost Process Group
Technology Enablers Group
Users' Group

Figure 4-25: Project organizational structure.

The Technology Enablers Group consists of representatives from all companies that provide technologies that enable the realization of the IDCT Tool; it includes both cost-valuation technologies and supporting infrastructure technologies. This group has two primary roles. The first is to facilitate the investigation of the technologies, in terms of its capabilities, application areas, methodologies utilized, etc. This usually includes providing the technology and documentation to the investigators over the span of the project, as well as providing training, technical support, and consultations. Their second role is to help us understand the design and cost-evaluation processes through their client organizations. One means to accomplish this is to provide the investigators training in the technology at client sites.

The Users' Group is composed of organizations, both from industry and academe that provide guidance from a user's perspective, as well as those that test and evaluate the Tool at various stages of development.

Table 4-7: Potential Follow-on Project Participants

Organization	**Expertise / *Software***
Acquisition Logistics Engineering (ALE) Columbus, OH Mr. Charles O. Coogan, President	Front-end analysis; life cycle costing; operations and support processes and analysis; systems engineering *Front End Trade Study Model, First Order LCC Model, etc*
Clemson University Department of Mathematical Sciences Dr. James A. Reneke, Professor	Affordability modeling and decision analysis
Cognition Corporation Bedford, MA Mr. Michael Cronin, CEO	Manufacturing cost modeling *Cost Advantage, Mechanical Advantage, Knowledge Center*
Frontier Technology, Inc. Dayton, OH Mr. Ronald D. Shroder, Director	Cost model integration *ICE (IDAPS Cost Estimation), WS-ICM (Weapons System Integrated Cost Model)*
Galorath Inc. El Segundo, CA Mr. Daniel Galorath, President	Parametric cost modeling and estimating *SEER Products: H, H/O&S, DFM, SEM*
James Gregory Associates Columbus, OH Dr. James Brink, President and CEO	Affordability assessment, integrated product and process development *PATA (Process Analysis Toolkit for Affordability*
Knowledge Base Engineering Centerville, OH Mr. E. I. "Sam" Nusinow, President	Object-oriented data base applications development; systems engineering *OZ (Object cZar)*
Knowledge Based Systems, Inc. (KBSI) College Station, TX Dr. Richard J. Mayer, President	Process modeling, cost modeling, activity-based costing *SmartCost, AIO WIN, SmartER, ProSim/ProCap*
SBIREC (Small Business Innovation Research Engineering Companies), Inc. Drumright, OK Mr. John Michael Lee, President and CEO	Manufacturing processes; virtual manufacturing enterprises
SDRC (Structural Dynamics Research Corporation), Milford, OH Dr. Mohsen Rezayat, Technical Fellow	Enterprise product data management, CAE/CAD/CAM *Metaphase Enterprise, I-DEAS, MetaSlate*
StraTech, Inc. Centerville, OH Dr. Richard Thomas, President	Preliminary and conceptual design; IPPD; Affordability
Systran Federal Corporation Beavercreek, OH Dr. V. ("Nagu") Nagarajan, Vice President & Program Manager	CORBA software development for UNIX and Windows; distributed computing software, databases & application development for enterprises in C/C++ and Java; Web-based software development; *ORB_IT (Object Request Broker – Integration technology)*
Tecolote Research, Inc. Beavercreek, OH Mr. Harmon T. Withee	Life cycle cost modeling, integrated product teams *ACEIT (Automated Cost Estimating Integrated Tools)*
The DFV Group, Inc. Virginia Beach, VA Mr. Edwin B. Dean, President	Systems engineering; designing for value; cost modeling and analysis; quality function deployment (QFD)

4.6 Conclusions, Recommendations, and Lessons Learned

4.6.1 Conclusions

1. The cost-evaluation environment, especially during conceptual and preliminary design is often not well understood, disparate and highly cross functional, involves information from all phases of the product life cycle, is highly time sensitive (assessments need to be made as the design evolves, not after the fact), involves technologically non-homogeneous designs, is dynamic and uncertain, knowledge based, model centric, is highly concerned with data relevancy and currency, and involves many islands of technologies that need to be assimilated and synthesized.

2. The above characteristics of the design/cost-evaluation environment provide major challenges to the development of a comprehensive design decision support system. It also provides great opportunities to improve current practices and processes.

3. Despite the difficult problem, this project demonstrates the feasibility of defining, developing, and deploying an effective system that can provide cost evaluation support to the design trade study processes.

4. The critical foundation for the successful development and use of an IDCT system is a thorough understanding and definition of the design decision-making processes that involve cost evaluation, definition of the cost-evaluation processes that support design, and the relationships between the design and cost-evaluation processes.
   - It is through these processes that the needs of the stakeholders are identified.
   - The greatest opportunity for improvement is often at the interfaces between processes, i.e., through better communications, control, and management.
   - Design decisions should be based on information that is drawn from all applicable sources and programs and from throughout their life cycle; the models that are used in the decision-making processes should be based on the information that is available.

5. Real value can be added by the development of an IDCT Tool if, in addition to opening the design space, it provides a means to identify and understand *how* and *why* design variables influence cost. Similarly, an effective IDCT Tool should capture and utilize evolving design experiences as a means to learn from the experiences.

6. In order to produce affordable products, cost analyses need to be an integral part of conceptual design. To this end, cost analyses need to directly support and facilitate design decision-making (trade studies) processes. In order for this to effectively occur, cost analyses need to be:
   - timely (opportune) – provide feedback as the design evolves, not after the fact.
   - inclusive (comprehensive) – provide, as appropriate, measures of the product's cost that consider production and operations costs, direct and indirect costs, and recurring and non-recurring costs.
   - useful (relevant) – be sensitive to the needs of the users; e.g., provide feedback on the effect of programmatic variables on cost to the program manager and affect of design variables on cost to the designer.
   - representative (accurate) – based on the most applicable data, models, knowledge, etc.

- non-intrusive -- be integrated into work processes, provide effective user interfaces, facilitate maintenance.

7. The fundamental capabilities that are needed by an IDCT Tool in order to effectively support early design decisions include:
   - assimilating existing and emerging cost evaluation and supporting technologies (costing systems, models, databases, design guidance, etc.) and providing an open system for easily incorporating new technologies,
   - managing cost-evaluation information over the product life cycle,
   - utilizing the most recent and most appropriate data, models, etc. that are available and basing the choice of data, models, etc. on the type of design/programmatic information that is currently available,
   - capturing evolving design experiences for the purpose of reuse and learning, and
   - providing design guidance from a cost perspective, both on the cost to produce the product and the cost to use the product.

4.6.2 Recommendations

1) Continue development of an IDCT Tool to enhance the design trade-study processes by providing effective decision-support through in situ cost evaluation.

2) Continue the definition of basic trade study processes and cost-evaluation support processes that were begun in this study. Develop case study examples of the processes from industry. Therefore, as a foundation for development, the design decision-making processes must be defined and the needs of the stakeholders in those processes need should be identified. Similarly, the cost-evaluation processes must be defined and the technologies that are used to support design should be identified. Finally, the relationships between the design and cost-evaluation processes need to be defined -- the greatest opportunity for improvement is often at the interfaces between processes, i.e., through better communications, control, and management.

3) Validate the characteristics of cost evaluation environment and general needs that were formulated in this study through an industry survey.

4) Validate the requirements for an effective in situ design cost evaluation system that were formulated in this study through an industry survey. From the survey, establish a industry-based prioritized list of requirements and an assessment of the risk of fulfilling the requirements.

4.6.3 Lessons Learned

- Specific user needs, and hence detailed IDCT system requirements, can only be defined once the applicable design trade study processes and support cost evaluation processes are defined.
- The development process, as evidenced through our experience with the prototype, is highly uncertain when using state-of-the-art technologies, even when using products from market leaders.

4.7 References

Blanchard, Benjamin S. and Walter J. Fabrycky. Systems Engineering and Analysis New Jersey: Prentice Hall. 1998.

Dant, Bob and Steve Kensinger. “Re-Engineering Engineering *Computer-Aided Engineering* March 1997: 60-63 and April 1997: pg. 56-60.

Dean, E. “Design for Competitive Advantage,” NASA Langley Research Center http://mijuno.larc.nasa.gov/, 1998.

Garvin, D. A. “What Does ‘Product Quality’ Really Mean?” *Sloan Management Review*, Fall 1984, pp. 34-52.

Greenwood, A. G. “A Framework for Manufacturing-Oriented, Design-Directed Cost Estimation,” Air Force Office of Scientific Research, Bolling Air Force Base, DC, September 1996.

Hisakazu, S., Yasuhiko, K., and Yuritsugu, T., “Using the Demanded Quality Deployment Chart: Developing the Quality Plan,” *Quality Function Deployment, Integrating Customer Requirements into Product Design*, Akao, J., Editor, Productivity Press, 1988.

Fondon, J. W. Jr., D. Rasmussen, D. Lum, and S. Smith, “Large Composite Structure -- Commercial Military Integration (Military Products Using Best Commercial Practices) Phase I -- Manufacturing 2005 Business Practices, Manufacturing Infrastructure, Process Technology Development, Volume II -- Manufacturing 2005 Handbook,” Manufacturing Technology Directorate, Wright Laboratory, Air Force Materiel Command, Wright-Patterson Air Force Base, OH January 1996.

Magrab, Edward B. *Integrated Product and Process Design and Development*. Boca Raton: CRC Press. 1997.

Mistree, F., W. Smith, B. Bras, J. K. Allen, and D. Muster, “Decision-Based Design: A Contemporary Paradigm for Ship Design,” *Transactions, Society of Naval Architects and Marine Engineers*, 1990, pp. 565-597.

Pugh, Stuart. *Total Design: Integrated Methods for Successful Product Engineering*. Wokingham, England: Addison-Wesley Publishing Company. 1991.

Rouse, William B. *Design for Success: A Human-Centered Approach to Designing Successful Products and Systems*. New York: John Wiley & Sons, Inc. 1991.

Rummler, G. A. and A. P. Brache, Improving Performance, 2nd Edition, San Francisco: Jossey-Bass Publishers, 1995.

Thomas, R. “Manufacturing Cost Estimation, Modeling and Control: A Review of the Literature,” Stratech, Inc., Beavercreek, OH August, 1994.

Ulrich, Karl T. and Steven D. Eppinger. *Product Design and Development*. New York: McGraw-Hill, Inc. 1995.

Vernadat, F. B. *Enterprise Modeling and Integration: Principles and Applications*. London: Chapman & Hall, 1996.

Yoshimura, M. “Concurrent optimization of product design and manufacture,” in *Concurrent Engineering: Contemporary issues and modern design tools*, H. R. Parsaei, Ed. London: Chapman & Hall, 1993, pp. 159-183.

4.8 Section 4 APPENDIXES

A. Product Design Methodology Definitions

B. Bibliography

C. IDL Specifications for Prototype

4.8.1 Appendix 4-A: Product Design Methodology Definitions

4.8.1.1 Ulrich & Eppinger

Source: Ulrich, Karl T. and Steven D. Eppinger. *Product Design and Development*. New York: McGraw-Hill, Inc. 1995.

1. Concept Development

   In the concept development phase, the needs of the target market are identified, alternative product concepts are generated and evaluated, and a single concept is selected for further development. A concept is a description of the form, function, and features of a product and is usually accompanied by a set of specifications, an analysis of competitive products, and an economic justification of the project.

   a) Identify Needs of Customer

      The goal of this activity is to understand customers' needs and to effectively communicate them to the development team. The output of this step is a set of carefully constructed customer need statements, organized in a hierarchical list, with importance weightings for each need.

   b) Establish Target Specifications

      Specifications are a precise description of what a product has to do. They are the translation of the customer needs into technical terms. Targets for the specifications are set early in the process and represent the hopes of the development team. Later these specifications are refined to be consistent with the constraints imposed by the team's choice of a product concept. The output of this stage is a list of specifications. Each specification consists of a metric and a target value for that metric.

   c) Analysis of Competitive Products

      An understanding of competitive products is critical to successful positioning of a new product and can provide a rich source of ideas for the product and production process design. Analysis of competitive products is also called competitive benchmarking. Competitive benchmarking is performed in support of the specification activity as well as in support of concept generation and concept selection.

   d) Generate and Evaluate Alternative Product Concepts

      The goal of concept generation is to explore thoroughly the space of product concepts that may be applied to meeting the customer needs. Concept generation includes a mix of external search, creative problem solving within the team, and systematic exploration of the various solution fragments the team generates. The result of this activity is usually a set of 10 to 20 concepts, each typically represented by a sketch and brief descriptive text.

   e) Select Single Concept for Further Development

      Concept selection is the activity in which various product concepts are analyzed and sequentially eliminated to identify one preferred concept. The process usually requires several iterations and may initiate additional concept generation and refinement.

   f) Refine Specification

      The target specifications set earlier in the process are revisited after a concept has been selected. At this point, the team must commit to specific values of the metrics reflecting the constraints inherent in the product concept, limitations identified through technical modeling, and trade-offs between cost and performance.

   g) Economic Analysis

      The team, often with the support of a financial analyst, builds an economic model for the new product. This model is used to justify continuation of the overall development program and to resolve specific trade-offs among, for example, development costs and manufacturing costs. While economic analysis is shown as one of the later activities in the concept development phase, an early economic analysis will almost always be performed before the project even begins.

h) Project Planning

In this final activity of concept development, the team creates a detailed development schedule, devises a strategy to minimize development time, and identifies the resources required to complete the project. The major results of the front-end activities can be usefully captured in a contract book that contains the mission statement, the customer needs, the details of the selected concept, the product specifications, the economic analysis of the product, the development schedule, the project staffing, and the budget. The contract book serves to document the agreement between the team and the senior management of the enterprise.

2. System Level Design

The system-level design phase includes the definition of the product architecture and the division of the product into subsystems and components. The final assembly scheme for the production system is usually defined during this phase as well.

a) Definition of Product Architecture

A product can be thought of in functional and physical terms. The functional elements of a product are the individual operations and transformations that contribute to the overall performance of the product. The physical elements of a product are the parts, components, and subassemblies that ultimately implement the product's functions. The physical elements of a product are typically organized into several major physical building blocks, called chunks. The architecture of a product is the scheme by which the functional elements of the product are arranged into physical chunks and by which the chunks interact.

b) Division of Product into Subsystem

This is the definition of how a product is going to be broken up into chunks.

c) Define Final Assembly Scheme

This is the definition of how the chunks are going to interact and be arranged.

d) Outputs: "Layout" of Product

The output of this phase is usually a geometric "layout" of the product, a functional specification of each of the product's subsystems, and a preliminary process flow diagram for the final assembly process.

3. Detailed Design

a) Complete Specifications

The detail design phase includes the complete specification of the geometry, materials, and tolerances of all of the unique parts in the product and the identification of all of the standard parts to be purchased from suppliers.

b) Establish Process Plan and Tooling

A process plan is established and tooling is designed for each part to be fabricated within the production system.

c) Output: Control Documentation

The output of this phase is the control documentation for the product – the drawings or computer files describing the geometry of each part and its production tooling, the specifications of the purchased parts, and the process plans for the fabrication and assembly of the product.

4. Testing and Refinement

The testing and refinement phase involves the construction and evaluation of multiple preproduction versions of the product.

a) Alpha Prototype

Early (alpha) prototypes are usually built with production-intent parts – parts with the same geometry and material properties as intended for the production version of the product but not necessarily fabricated with the actual processes to be used in production. Alpha prototypes are tested to determine whether or not the product will work as designed and whether or not the product satisfies the key customer needs.

b) Beta Prototype

Later (beta) prototypes are usually built with parts supplied by the intended production processes but may not be assembled using the intended final assembly process. Beta prototypes are extensively evaluated internally and are also typically tested by customers in their own use environment. The goal for the beta prototypes is usually to answer questions about performance and reliability in order to identify necessary changes for the final product.

5. Production Ramp-Up

In the production ramp-up phase the product is made using the intended production system. The artifacts produced during production ramp-up are sometimes supplied to preferred customers and are carefully evaluated to identify any remaining flaws.

a) Train Work Force

The purpose of the ramp-up is to train the work force and to work out any remaining problems in the production processes.

b) Transition into Production

The transition from production ramp-up to ongoing production is usually gradual and continuous. At some point in this transition, the product is launched and becomes available for widespread distribution.

4.8.1.2 Magrab

Source: Magrab, Edward B. *Integrated Product and Process Design and Development*. Boca Raton: CRC Press. 1997. Methodology is shown on a flow chart, pg. 36.

1. Establish Company Strategy

What are the product's goals, customers, and benefits? What is the business plan for the new product? What is the reason for this product? What does the organization hope to gain by developing such a product? Will it help achieve the organization's long term goals?

a) Establish Cross-Functional Team

The team should include members from engineering, production, finance, marketing, and purchasing to provide the input necessary for a proven product. It may even include representatives from key suppliers.

2. Establish Customer Needs

Determine what the customer wants.

3. Define Product

Translate the needs of the customer into specific attributes of the product.

a) Determine Feasibility of Product and Compare Strategy

Perform a cost analysis on the product to see if it is worth the company's time and beneficial for the company.

b) Competition Identified and Benchmarked

Determine what is available on the market to assess what attributes the product needs and the level to which the product concept compares to what is already available.

c) Adjust Product Definition as Required and Draw Up Preliminary Product Specifications

What are the specific features and performance issues?

4. Generate Feasible Designs

Generate feasible designs through brainstorming, etc.

5. Evaluate Feasible Designs

Compare the possible design to each other to determine which concept is the most feasible.

a) Explore the Most Promising Concepts and Try Different Configurations

Use the most promising concepts in different configurations to find out if better total concepts exist.

b) Choose Best Configuration

c) Generate Drawings and Prototypes

Using the chosen configuration, proceed with the total final design.

d) Evaluate Prototypes Compared to Preliminary Product Specifications

Test the product to determine as to whether the product meets the customer's needs or not.

6. Process Design

   a) Design Manufacturing Processes and Assembly Procedures

7. Manufacture and Assemble
8. Market Product

4.8.1.3 Pugh

Source: Pugh, Stuart. *Total Design: Integrated Methods for Successful Product Engineering*. Wokingham, England: Addison-Wesley Publishing Company. 1991.

1. Examine Market

   The starting point for any design should be the establishment of the market/user need situation in considerable depth. It is common practice to produce a device or document, usually referred to as a 'brief', at this stage. No matter how well designed a product is, it will be a failure if it does not sell.

2. Specifications

   Specifications are drawn up that will cause the product to meet the needs of the market. From these specifications, all actions regarding the product should radiate. It becomes the frame of reference for all future decisions. Although the specifications are not static, good reasoning must be used when changing the specifications and the end result should bear strong resemblance to the initial specifications.

3. Concept Design

   This stage's primary concern is generating a set of possible, system-level solutions that meet the stated specifications.

   a) Generate Solutions to Meet Stated Need

   Using a variety of techniques, generate possible solutions that meet the specifications laid forth in the previous step. Be sure to explore several different avenues.

   b) Evaluate Solutions

   Criteria need to be determined to effectively evaluate the concepts. These criteria should come from the specifications. This process should be iterative, where new possible concepts evolve from the converging effect of eliminating concepts that do not meet the specifications.

4. Detailed Design

   This stage is where the final, chosen concept is completely designed and tested.

5. Manufacture

   This stage should actually begin during the Detailed Design phase so that the final product can easily be manufactured and transitioned into production. It is where the product design is produced for the market.

6. Sell

   This stage is the marketing of the final product – distribution, service back-up, etc., which are all part of the marketing or selling activity.

4.8.1.4 Rouse

Source: Rouse, William B. *Design for Success: A Human-Centered Approach to Designing Successful Products and Systems*. New York: John Wiley & Sons, Inc. 1991.

Rouse only presents simplified design processes to use with a system of measurements that ensure human-centeredness. The focus of this work is designing products and systems that are market-driven and user-oriented. This product design process is general and easily defined given a foreknowledge of product design.

1. Recognition
2. Formulation
3. Analysis
4. Synthesis
5. Fabrication

4.8.1.5 Dant & Kensinger

Source: Dant, Bob and Steve Kensinger. "Re-Engineering Engineering," *Computer-Aided Engineering* March 1997: 60-63 and April 1997: pg. 56-60.

In this work, the authors share a four-step design procedure. However, they only define the sub-steps of the Development Phase. The other three phases names imply what they are about, but only the second phase is defined with any detail.

1. Innovation Stage
2. Development Stage

   The Development Stage is a series of links or loops leading from the Innovation Stage to the Production Stage.

   a) Incubation Phase

      Incubation is where ideas are heard and formulated into products. An integrated-discipline team from marketing, engineering, manufacturing, finance, and distribution reviews these ideas. This panel of experts must have the capability to review ideas, synthesize concepts, and create preliminary designs and production scenarios. Once they have reviewed the product, attached value to its assets, and determined its return on investment, they make a go or no-go decision regarding the product. If the team finds that a product has enough value to move forward into the next phase, it establishes a "product control packet." This packet is the high-level control plan for a product and becomes a living document for the life cycle of a product. It is made up of a market plan, operating plan, business plan, and product specification.

   b) Market and Research Phase

      In this phase, the team's primary purpose is defining and understanding the product's value chain – the functional requirements. The key aspect of this activity is the relationship between the market and the function of the product. Once the product is understood, marketing is responsible for the market plan, finance for the business plan, operations for the manufacturing and operating plans, and engineering for the design plan and product specifications. The control packet now contains well-defined measures and goals for the product, which are reviewed at each product review filter.

c) Engineering Phase

During the Engineering Phase, engineers must develop options that meet the functional requirements outlined in the product specifications – and fit those into the geometric, material, power, and other envelopes defined by the product specification. The team reviews these options, and the most cost-effective solution is designed. Design prototypes are also developed.

d) Product Review Filter

Between each phase resides a "product control link," or loop, which contains a "product review filter." The product control loop is established for each product. It compares current actual data to previous assumptions or projections from previous phases and facilitates flow of information in and out of the current phase. The product review filter is in place to ensure that all control aspects of the product are reviewed before the product moves into the next phase.

e) Documentation Phase

This phase closes the circle on the three "F's" of engineering – function, fit, and form. The bases for the Documentation Phase are functional attributes, consisting of acceptable envelopes for geometry and other design characteristics critical to satisfaction of the design specification. The documentation phase is where CAD/CAM tools become most effective. As the form is finalized, its databases join the other active product information in the Product Notebook, a storing place for component specifications. The product is documented for the most cost-effective manufacturing effort that includes inspection and certification.

f) Procurement Phase

The Product Notebook is now complete. When supplier qualifications are completed, initial purchase orders are placed and manufacturing schedules are defined. The Procurement Phase is where the day-to-day production effort is defined and controlled. Purchasing will search out the most cost-effective suppliers and material that completely satisfies the specifications defined in the previous phases.

g) Distribution Phase

Once the product passes into the Distribution Phase, final packaging, order entry systems, logistics, warehousing, and quality assurance procedures are finalized.

h) Product Launch Phase

With a last pass through the final Product Review Filter, the Integrated Discipline Team is ready for the product to enter mass production, selling, and distribution. To turn the product over to the company, the team must educate sales, manufacturing, distribution, quality assurance, advertising, and accounting regarding the product's value chain.

3. Production
4. Obsolescence

4.8.1.6 Blanchard & Fabrycky

Source: Blanchard, Benjamin S. and Walter J. Fabrycky. *Systems Engineering and Analysis* New Jersey: Prentice Hall. 1998.

This book deals with the design of a system, but its application is very applicable to a product, especially since a product can be viewed as a system. A representation of this methodology is presented on page 20. Steps 4, 5, and 6 are touched on only briefly.

1. Conceptual Design

Conceptual design is the foundation on which the life-cycle phases of preliminary system design, detail design and development, and so forth are based. Within the conceptual design phase are activities related to the identification of customer need and the several steps involved in the definition of system design requirements. It converts the customer needs to system design criteria.

g) Definition of Need

The design process begins with the identification of a "need," "want," or "desire" for one or more new entities, or for a new or improved capability. It should be based on a real (or perceived) deficiency.

h) Feasibility Study

Having a definition of need, it is then necessary to (1) identify possible system-level design approaches that can be pursued to meet that need; (2) evaluate the most likely approaches in terms of performance, effectiveness, maintenance and logistic support, and economic criteria; and (3) recommend a preferred course of action. There may be many possible alternatives; however, the number must be narrowed down to a few feasible ones, consistent with the availability of resources (i.e. personnel, materials, and money).

i) Advance Product Planning

Given an identified need for a feasible new system, advance planning activities are initiated that will establish a project for the conceptual design of that system. These include (1) communication with the customer (consumer) to obtain an in-depth delineation of the need, (2) completion of a feasibility analysis to determine the availability of technologies applicable to the need, (3) definition of system operational requirements, (4) development of the system maintenance and support concept, (5) identification and prioritization of technical performance measures, (6) completion of a top-level functional analysis, (7) preparation of a system specification, and (8) conduct of a conceptual design review.

2. Preliminary Design

Preliminary system design begins by establishing a functional baseline for the system. It extends the translation of system-level requirements into design requirements for the subsystem level, configuration-item level, and below.

a) System Functional Analysis

Functional Analysis is the iterative process of breaking down (or decomposing) requirements from the system level, to the subsystem level, and as far down the hierarchical system structure as necessary to describe function interfaces and identify resource needs adequately.

b) Preliminary Synthesis and Allocation of Design Criteria

Performance factors, design factors, and effectiveness requirements need to be assigned. The system support requirements also need to be allocated.

c) System Optimization

A study of system and subsystem trade-offs needs to occur, along with an evaluation of alternatives. This ensures the best solution for the design of the system.

d) System Synthesis and Definition

System synthesis is achieved when sufficient preliminary design progress has been made and trade-off studies have been accomplished to confirm and assure the completeness of system performance and other design requirements. The performance, configuration, and arrangement of a chosen system and its elements are portrayed along with the techniques for their test, operation, and life-cycle support. Those portrayals cover intrasystem and intersystem and item interfaces, and provide enough information forming a definitive baseline that can be presented in the form of detail specifications.

3. Detail Design

With the definition of the overall system and its major subsystems in hand, one may proceed to the realization of specific system components. Realization includes the technical activities of (1) describing subsystems, units, assemblies, lower-level components, software, people, and elements of logistic support that make up the system; (2) preparing specifications and design data for all system components; (3) procuring detail design for unique items; (4) developing an engineering model, a service test model, or a prototype model of the system and its components for test and evaluation; (5) integrating system components and testing the integration to verify compliance with the specified requirements; and (6) redesigning, reengineering, and retesting the system or an element thereof.

a) System-Product Design

The system-product design involves a detail design of the functional systems (prime equipment and software), system maintenance and logistic support elements, support functions, data and documentation, analysis and evaluation, and design review.

b) System Prototype Development

A model is developed to test the system outside of the theoretical realm.

c) System Prototype Test and Evaluation

This phase involves preparing a prototype test, actually testing the system and equipment, analyzing the data and evaluating the data, reporting the test data, and making modifications as necessary.

4. Production and/or Construction

The detailed design is actually taken and manufactured, installed, and implemented. Training may be necessary in this phase.

a) System Evaluation

After the system has been produced, an evaluation into how successful the system meets the needs of the customer is performed.

b) Modification for Corrective Action and/or Product Improvement

If the system does not meet the needs of the customer, modifications should be implemented into the system. If the customer's needs change, the system can be improved.

5. Utilization and Support

Future support of the system should be an on-going task for the producer of the system.

a) System Evaluation, Analysis, and Evaluation

Same ideas as 4a.

b) Modification for Corrective Action and/or Product Improvement

Same idea as 4b.

6. Phaseout

In the future, the technology and/or services involved with the system will become obsolete. At this point, the system should be phased out and replaced with a newer system if necessary.

4.8.1.7 Composite

The following is our compilation of the product development activities that are derived from the six methodologies that are defined above. The sources of the activities are noted in parentheses.

1. Establish company strategy (Magrab)

What are the product's goals, customers, and benefits? What is the business plan for the new product? What is the reason for this product? What does the organization hope to gain by developing such a product? Will it help achieve the organization's long term goals?

a) Establish cross-functional team (Magrab)

The team should include members from engineering, production, finance, marketing, and purchasing to provide the input necessary for a proven product. It may even include representatives from key suppliers.

2. Establish customer needs (U&E, Magrab, Pugh, D&K, B&F)

The starting point for any design should be the establishment of the market/user need situation in considerable depth. It is common practice to produce a device or document, usually referred to as a "brief" or "control packet," at this stage. No matter how well designed a product is, it will be a failure if it does not sell. Specifications are drawn up that will cause the product to meet the needs of the market. From these specifications, all actions regarding the product should radiate. It becomes the frame of reference for all future decisions. Although the specifications are not static, good reasoning must be used when changing the specifications and the end result should bear strong resemblance to the initial specifications.

This step entails a comprehensive study of the customers and markets that the product is intended to be purchased by and appeal to. Interviews, surveys, etc. should be completed that show what the market/customer wants out of the product. These "wants" become translated to specifications.

i) Establish Target Specifications

Specifications are a precise description of what a product has to do. They are the translation of the customer needs into technical terms. Targets for the specifications are set early in the process and represent the hopes of the development team. Later these specifications are refined to be consistent with the constraints imposed by the team's choice of a product concept. The output of this stage is a list of specifications. Each specification consists of a metric and a target value for that metric.

3. Define product (Magrab)

Translate the needs of the customer into specific attributes of the product.

a) Determine feasibility of product and compare to strategy (Magrab)

Perform a cost analysis on the product to see if it is worth the company's time and beneficial for the company.

b) Prepare list of metrics (U&E 57)

These metrics are measurements of product performance and capability. They are drawn from the wants and desire of the customer.

c) Competition identified and benchmarked (Magrab, U&E 61)

An understanding of competitive products is critical to successful positioning of a new product and can provide a rich source of ideas for the product and production process design. Analysis of competitive products is also called competitive benchmarking. Competitive benchmarking is performed in support of the specification activity as well as in support of concept generation and concept selection.

d) Set ideal and marginally acceptable target values for each metric (U&E 61, Magrab, Pugh)

After research of the product and competitive benchmarking similar products has been completed, ideal and marginally acceptable target values for each metric should be defined so that goals can be set for the future design.

e) Reflect on the results (U&E 65)

The previous steps need to be reviewed so that lessons can be learned for future application and the "control packet" is updated to provide documentation of the steps taken to date.

4. Concept Development (Pugh, D&K, B&F)

In the concept development phase, alternative product concepts are generated and evaluated, and a single concept is selected for further development. A concept is a description of the form, function, and features of a product and is usually accompanied by a set of specifications, an analysis of competitive products, and an economic justification of the project.

a) Generate Feasible Designs (Magrab, Pugh, D&K, B&F)

The goal of concept generation is to explore thoroughly the space of product concepts that may be applied to meeting the customer needs. Concept generation includes a mix of external search, creative problem solving within the team, and systematic exploration of the various solution fragments the team generates. The result of this activity is usually a set of 10 to 20 concepts, each typically represented by a sketch and brief descriptive text.

b) Evaluate Feasible Designs (Magrab, Pugh, D&K)

Compare the possible designs to each other to determine which concepts are the most feasible.

c) Explore the most promising concepts and try different configurations (Magrab)

Use the most promising concepts in different configurations to find out if better total concepts exist.

d) Select single concept for further development (U&E, Magrab)

Concept selection is the activity in which various product concepts are analyzed and sequentially eliminated to identify one preferred concept. The process usually requires several iterations and may initiate additional concept generation and refinement.

e) Refine Specifications (U&E)

The target specifications set earlier in the process are revisited after a concept has been selected. At this point, the team must commit to specific values of the metrics reflecting the constraints inherent in the product concept, limitations identified through technical modeling, and trade-offs between cost and performance.

5. Market and Research Phase (D&K)

In this phase, the team's primary purpose is defining and understanding the product's value chain – the functional requirements. The key aspect of this activity is the relationship between the market and the function of the product. Once the product is understood, marketing is responsible for the market plan, finance for the business plan, operations for the manufacturing and operating plans, and engineering for the design plan and product specifications. The control packet now contains well-defined measures and goals for the product, which are reviewed at each product review filter.

a) Feasibility/Economic study (U&E 233)

The team, often with the support of a financial analyst, builds an economic model for the new product. This model is used to justify continuation of the overall development program and to resolve specific trade-offs among, for example, development costs and manufacturing costs. While economic analysis is shown as one of the later activities in the concept development phase, an early economic analysis will almost always be performed before the project even begins (see step 3a).

b) Project Planning (U&E 259, B&F)

In this final activity of concept development, the team creates a detailed development schedule, devises a strategy to minimize development time, and identifies the resources required to complete the project. The major results of the front-end activities can be usefully captured in a contract book that contains the mission statement, the customer needs, the details of the selected concept, the product specifications, the economic analysis of the product, the development schedule, the project staffing, and the budget. The contract book serves to document the agreement between the team and the senior management of the enterprise.

6. System Level Design (U&E, Magrab, B&F)

The system-level design phase includes the definition of the product architecture and the division of the product into subsystems and components. The final assembly scheme for the production system is usually defined during this phase as well. Preliminary system design begins by establishing a functional baseline for the system. It extends the translation of system-level requirements into design requirements for the subsystem level, configuration-item level, and below.

a) Def. of Product Architecture (U&E 129)

A product can be thought of in functional and physical terms. The functional elements of a product are the individual operations and transformations that contribute to the overall performance of the product. The physical elements of a product are the parts, components, and subassemblies that ultimately implement the product's functions. The physical elements of a product are typically organized into several major physical building blocks, called chunks. The architecture of a product is the scheme by which the functional elements of the product are arranged into physical chunks and by which the chunks interact.

1) Division of product into subsystems (U&E)
   This is the definition of how a product is going to be broken up into chunks.
2) Define final assembly scheme (U&E)
   This is the definition of how the chunks are going to interact and be arranged.

b) System functional analysis (B&F)

Functional Analysis is the iterative process of breaking down (or decomposing) requirements from the system level, to the subsystem level, and as far down the hierarchical system structure as necessary to describe function interfaces and identify resource needs adequately.

c) Preliminary synthesis and alleviation of design criteria (B&F)

Performance factors, design factors, and effectiveness requirements need to be assigned. The system support requirements also need to be allocated.

d) System Optimization (B&F)

A study of system and subsystem trade-offs needs to occur, along with an evaluation of alternatives. This ensures the best solution for the design of the system.

e) System Synthesis and Definition (B&F)

System synthesis is achieved when sufficient preliminary design progress as been made and trade-off studies have been accomplished to confirm and assure the completeness of system performance and other design requirements. The performance, configuration, and arrangement of a chosen system and its elements are portrayed along with the techniques for their test, operation, and life-cycle support. Those portrayals cover intrasystem and intersystem and item interfaces, and provide enough information forming a definitive baseline that can be presented in the form of detail specifications.

f) Output: "layout" of product (U&E)

The output of this phase is usually a geometric "layout" of the product, a functional specification of each of the product's subsystems, and a preliminary process flow diagram for the final assembly process.

7. Detail Design (U&E, Pugh, D&K, B&F)

With the definition of the overall system and its major subsystems in hand, one may proceed to the realization of specific system components. Realization includes the technical activities of (1) describing subsystems, units, assemblies, lower-level components, software, people, and elements of logistic support that make up the system; (2) preparing specifications and design data for all system components; (3) procuring detail design for unique items; (4) developing an engineering model, a service test model, or a prototype model of the system and its components for test and evaluation; (5) integrating system components and testing the integration to verify compliance with the specified requirements; and (6) redesigning, reengineering, and retesting the system or an element thereof.

a) Complete Specifications (U&E, B&F)

The detail design phase includes the complete specification of the geometry, materials, and tolerances of all of the unique parts in the product and the identification of all of the standard parts to be purchased from suppliers.

b) Establish Process Plan and Tooling (U&E 179)

A production process plan is established and tooling is designed for each part to be fabricated within the production system.

c) Output: Control Documentation (U&E, D&K)

The output of this phase is the control documentation for the product – the drawings or computer files describing the geometry of each part and its production tooling, the specifications of the purchased parts, and the production process plans for the fabrication and assembly of the product.

d) System Prototype Development (U&E, Magrab, B&F)

1) Alpha Prototype

Early (alpha) prototypes are usually built with production-intent parts – parts with the same geometry and material properties as intended for the production version of the product but not necessarily fabricated with the actual processes to be used in production. Alpha prototypes are tested to determine whether or not the product will work as designed and whether or not the product satisfies the key customer needs.

2) Beta Prototype

Later (beta) prototypes are usually built with parts supplied by the intended production processes but may not be assembled using the intended final assembly process. Beta prototypes are extensively evaluated internally and are also typically tested by customers in their own use environment. The goal for the beta prototypes is usually to answer questions about performance and reliability in order to identify necessary changes for the final product.

e) System Prototype Test and Evaluation (U&E, Magrab, B&F)

This phase involves preparing a prototype test, actually testing the system and equipment, analyzing the data and evaluating the data, reporting the test data, and making modifications as necessary.

8. Process Design (Magrab)

a) Design manufacturing processes and assembly procedures

9. Procurement phase (D&K)

The Product Notebook is now complete. When supplier qualifications are completed, initial purchase orders are placed and manufacturing schedules are defined. The Procurement Phase is where the day-to-day production effort is defined and controlled. Purchasing will search out the most cost-effective suppliers and material that completely satisfies the specifications defined in the previous phases.

10. Distribution Phase (D&K)

Once the product passes into the Distribution Phase, final packaging, order entry systems, logistics, warehousing, and quality assurance procedures are finalized.

11. Production Ramp-Up (U&E, D&K)

In the production ramp-up phase, the product is made using the intended production system. The artifacts produced during production ramp-up are sometimes supplied to preferred customers and are carefully evaluated to identify any remaining flaws.

a) Train Work Force (U&E)

The purpose of the ramp-up is to train the work force and to work out any remaining problems in the production processes.

b) Transition Into Production (U&E, Magrab, Pugh, D&K, B&F)

The transition from production ramp-up to ongoing production is usually gradual and continuous. At some point in this transition, the product is launched and becomes available for widespread distribution.

12. Production (U&E, Magrab, Pugh, D&K, B&F)

This stage should actually begin during the Detailed Design phase so that the final product can easily be manufactured and transitioned into production. It is where the product design is produced for the market.

a) System Assessment (B&F)

After the system has been produced, an assessment into how successful the system meets the needs of the customer is performed.

b) Modification for Corrective Action and/or Product Improvement (B&F)

If the system does not meet the needs of the customer, modifications should be implemented into the system. If the customer's needs change, the system can be improved.

13. Utilization and Support (B&F)

a) System Assessment, Analysis, and Evaluation (B&F)

After the system has been produced, an assessment into how successful the system meets the needs of the customer is performed.

b) Modification for Corrective Action and/or for Product Development (B&F)

If the system does not meet the needs of the customer, modifications should be implemented into the system. If the customer's needs change, the system can be improved.

14. Sell (Pugh, Magrab)

This stage is the marketing of the final product – distribution, service back-up, etc., which are all part of the marketing or selling activity.

15. Phaseout (D&K, B&F)

In the future, the technology and/or services involved with the system will become obsolete. At this point, the system should be phased out and replaced with a newer system if necessary.

4.8.2 Appendix 4-B: Biblography

4.8.2.1 Activity-Based Costing

Brimson, J. A. (1999). Feature Costing; Beyond ABC. Emerging Practices in Cost Management**:** B1-1 - B1-8.

Erraguntla, M., P. Benjamin, et al. (1998). Using SmartABC for Integrating Activity Based Costing with Activity Modeling. College Station, TX, Knowledge Based Systems, Inc.**:** 21.

Roztocki, N. and K. L. Needy (1999). "Integrating Activity-Based Costing and Economic Value Added in Manufacturing." Engineering Management Journal **11**(June 1999): 17-22.

Yoshikawa, T., J. Innes, et al. (1994). Functional Analysis of Activity-based Cost Informantion. Activity Based Management. B. J. Brinker, Warren, Gorham, and Lamont**:** 13-1 - 13-9.

4.8.2.2 Cost Estimating Models and Tools

Boothroyd, G. and C. Reynolds "Approximate Cost Estimates for Typical Turned Parts." Journal of Manufacturing Systems **8**(3): 185-193.

Collins, J. and D. Eck (1999). Parametric Estimating Handbook, Department of Defense**.**

Eskew, H. L. and K. Lawler, S. (1994). A Study of the Predictive Accuracy of Alternatively Estimated Statistical Cost Models. Alexandria, VA, Center for Naval Analysis, Professional Paper 535.

Eskew, H. L. and W. R. Nunn (1995). Procedures and Software for Assessing Uncertainty in Cost Estimates. Alexandria, VA, Center for Naval Research, Report No. CRM 95-87.

Frazier, T. P. and T. R. Gulledge, Estimating Fixed and Variable Costs for Aircraft Manufacturers, ICA/NES Joint National Meeting, Huntsville, AL, June 1-3, 1998

Gutowski, T., D. Hoult, et al. (1994). "Development of a Theoretical Cost Model for Advanced Composite Fabrication." Composites Manufacturing.

Hanson, C. L. (1989). An Evaluation of the Automated Cost Estimating Integrated Tools (ACEIT) System. Wright Patterson Air Force Base, OH, Air Force Institute of Technology, Thesis.

Hazelrigg, G. A. On the Role and Use of Mathematical Models in Engineering Design. Arlington, VA, National Science Foundation**:** 11.

Hodson, M. and M. Matthews (1997). A Comprehensive Cost Model for Aerospace Manufacturing?, Business Process Resource Center, http://brpc.warwick.ac.uk/focus10.html.

Hoult, D. P. and L. C. Meador (1996). Predicting Product Manufacturing Costs from Design Attributes: A Complexity Theory Approach. International Congress & Exposition, Detroit, Michigan, SAE

International.

Ilcewicz, L. B., G. E. Mabson, et al. (1996). Cost Optimization Software for Transport Aircraft Design Evaluation (COSTADE) Design Cost Methods. Hampton, VA, NASA Langley Research Center.

Jha, N. K. (1992). “Stochastic Modelling and Manufacturing Cost Estimation in Uncertain Industrial Environments.” International Journal of Production Economics 30(12): 2755-2771.

Kankey, R. and J. Robbins, Eds. (1991). Cost Analysis and Estimating: Shifting U.S. Priorities. New York, Springer-Verlag.

Mabison, G. E., L. B. Ilcewicz, et al. (1996). Cost Optimization Software for Transport Aircraft Design Evaluation (COSTADE), NASA, Langley Research Center, Contractor Report 4736, Contracts NAS1-18889 and NAS1-20013.

Marx, W. J., D. N. Mavris, et al. (1995). A Hierarchial Aircraft Life Cycle Cost Analysis Model. 1st AIAA Aircraft Engineering, Technology, and Operations Congress, Los Angeles, CA, American Institute of Aeronautics and Astronautics.

Matthews, M. L., Productivity Analysis and Cost Modeling, http://www.anu.edu.au/mba/faculty/mlm/mlmprod.html, 1998

Noble, J. S. and J. M. A. Tanchoco Cost Modeling for Design Justification. Cost Anlaysis: Applications of Economics and Operations Research. T. R. Gulledge, Jr. and Litteral. New York, Springer-Verlag**:** 197-213.

Thomas, R. (1994). Manufacturing Cost Estimation, Modeling and Control: A Review of the Literature. Beavercreek, OH, Stratech, Inc.

Twomey, M. G. (1991). A Review of Selected USAF Life Cycle Costing Models. Air Force Institute of Technology. Wright-Patterson Air Force Base, OH, AFIT Thesis**:** 286+ix.

4.8.2.3 Cost Estimating Process

AIR FORCE COST ANALYSIS AGENCY, http://www.saffm.hq.af.mil/navbar/afcaabar.html#

Anderson, T. P. and J. S. Cherwonik (1997). “Cost Estimating Risk and Cost Estimating Uncertainty Guidelines.” Aquisition Review Quarterly(Summer 1997): 339-348.

Boons, A.N.A.M., Product Costing for Complex Manufacturing Systems, Int. J. Production Economics, v55, 241-255, 1998

Dean, E.B. and V.A. Hampton, *Genopersistating the System*, Aerospace Design Conference, Irvine, CA(AIAA 93-1031), February 16-19, 1993

Greenwood, A. G. (1997). An Approach to Enhance Cost Estimation During Product-Process Design.

Decision Sciences Institute Conference, San Diego.

Greenwood, A. G. (1996). A Framework for Manufacturing-Oriented, Design-Directed Cost Estimation. Bolling Air Force Base, DC, Air Force Office of Scientific Research.

Gulledge, T. R., W. P. Hutzler, et al., Eds. (1992). Cost estimating and Analysis: Balancing Technology and Declining Budgets. New York, Springer-Verlag.

Gulledge, T. R., Jr. and Litteral, Eds. Cost Anlaysis: Applications of Operations Research. New York, Springer-Verlag.

Humpheys, K. K. and P. Welman (1996). Basic Cost Engineering. New York, Marcel Dekker, Inc.

Ostwald, P. F. (1992). Engineering Cost Estimating. Englewood Cliffs, NJ, Prentice Hall.

Stewart, R. D. (1990). Cost Estimating. New York, Wiley.

Stewart, R. D. and R. M. Wyskida (1987). Cost Estimator's Reference. New York, John Wiley & Sons.

Vicinanza, S. S., M. J. Prietula, et al. (1990). Case-based software cost estimation. Pittsburgh, PA, Carnegie Mellon University.

4.8.2.4 Cost Management

Risk Management Concepts and Guidance, Defense Systems Management College.

Anthony, R. N. (1984). "Cost allocation." The Journal of Cost Analysis **1**(1): 5-15.

Ballou, P. O. (1980). "A Functional View of Organizations." Journal of Defense Systems Acquistion Management **3, #4**(Autumn, 1980): 102-112.

Blanchard, B. S. (1978). Design and Manage to Life Cycle Cost. Portland, OR, M/A Press.

Cochran, E. B. (1968). Planning Product Costs: Using the Improvement Curve. San Francisco, Chandler Publishing Company.

DuBrosky, B. M. and R. Walker (1997). Cost Integration Strategies for More Affordable Aerospace Designs. World Aviation Congress, Anaheim, CA, AIAA - American Institute of Aeronautics.

Fabrycky, W. J. and B. J. Blanchard (1991). Life-Cycle Cost and Economic Analysis. Englewood Cliffs, NJ, Prentice Hall.

Gindele, M. E. (1993). "Reengineering Cost Analysis: A Case for Cost Reform." Program Manager(November-December 1993): 26-29.

Greenberg, J. S. and G. A. Hazelrigg (1973). A Reliability, Cost, ande Risk Analysis of Establishing and Maintaining a Space Communications Satellite Network. AIAA/ASME/SAE Space Mission Planning and Exeution Meeting, Denver, CO, AIAA.

Kaminski, P. G. (1995). The Defense Acquisition Challege: Technological Supremacy at an Affordable Cost. JAST Industry Conference.

Kandebo, S. W., U.S. Initiative Aims to Cut Military Aircraft Costs, Aviation Week & Space Technology, 79-82 June 3, 1996

Li, M., E. Kendall, et al. A Computer System for Life-Cyle Cost Estimation and Manufacturability Assessment of Composites, Computer Systems Engineering - Royal Melbourne Insitute of Technology.

Moore, M. D., H. C. Loc, et al. (1997). Economic Optimization of an Advanced Subsoni Transport. World of Aviation Congress, Anaheim, CA.

`

Raffi, F. and P. M. Swamidass (1987). “Towards a Theory of Manufacturing Overhead Cost Behavior: A Conceptual and Empirical Analysis.” Journal of Operations Management **1+2**: 121-137.

Rush, B. C. (1997). “Cost as an Independent Variable: Concepts and Risks.” Aquisition Review Quarterly(Spring 1997): 161-172.

Yoshikawa, T. (1993). Contemporary cost management. London ; New York, Chapman & Hall University and Professional Division.

Dean, E. B. (1992). Elements of Designing for Cost. Aerospace Design Conference, Irvine, CA, American Institute of Aeronautics and Astronautics.

4.8.2.5 Design Process

Arnold, W. V. (1986). Designing Defense Systems: The Trade-Off Process. Fort Belvoir, VA, Defense Systems Management College.

Bailey, M. W. (2000). A Federated Intelligent Product Envionment. Cincinnati, OH, GE Aircraft Engines**.**

Bailey, M. W. (1999). Integrated Multidisciplinary Design. American Institute of Aeronautics and Astronautics. Cincinnati, OH, GE Aircraft Engines**:** 1-11.

Balabanov, V. O., A. A. Giunta, et al. (1999). “Reasonable Design Approach to Response Surface Approximation.” Journal of Aircraft 36 (January- February1999): 308-315.

Batill, S. M., M. A. Stemlack, et al. (1998). “Framework for Multidiciplinary Design Based on Response-Surface Approximations.” AIAA (August 1998): 287-297.

Battershell, A. L. (1995). “Technology Approach: DoD Versus Boeing (A Comparative Study).” Acquisition Review Quarterly(Summer 1995): 213-230.

Battershell, L. A. (1995). “Technology Approach: DoD Versus Boeing (A Comparative Study).” Acquisition Review Quarterly(Summer 1995): 213-230.

Beeby, W. D. (1983). “The Heart of Integration: Establishing a Computer-Accessible Set of Facts About Design and Manufacturing is a Major Undertaking with Potentially Great Rewards.” IEEE Spectrum(May 1983): 44-48.

Blanchard, B. S. and W. J. Fabrycky (1998). Systems Engineering and Analysis. Upper Saddle River, NJ, Prentice Hal.

Boothroyd, G. and P. Dewhurst (1988). “Product Design for Manufacture and Assembly.” Manufacturing Engineering(April 1988): 42-46.

Bossert, J. L. (1991). Quality Function Deployment. Milwaukee, WI, ASQC Quality Press.

Bras, B. and F. Mistree (1991). Designing Design Processes in Decision-Based Concurrent Engineering. Aerospace Product/Process Design Interface. Warrendale, PA, Society of Automotive Engineers, Inc.

Chen, C.-S., J. K. Chow, et al. (1993). Concurrent Product Design Evaluation and Decision Support. Concurrent Engineering: Methodology and Applications. P. Gu and A. Kusiak. NewYork, Elsevier**:** 235-247.

Cutler, J. (1981). Understanding Aircraft Structures. London, Granada.

Dant, Bob and Steve Kensinger. “Re-Engineering Engineering *Computer-Aided Engineering* March 1997: 60-63 and April 1997: pg. 56-60.

Dubensky, R. G., D. E. Jay, et al. (1989). Process driven design of a plastic bumper beam. Fifth Annual ASM/ESD Advanced Composites Conference, Dearborn, MI, ASM International.

El Aichaoui, S., M. Hale, et al. (1998). Building Design Applications Using Process Elements. 7th AIAA/USAF/NASA/ISSMO Symposium on Multidiciplanary Analysis and Optimization, St Louis, MO, AIAA.

Gilman, C., B. J. Aparicio, et al. (1997). “Integration of design and manufacturing in a virtual enterprise using enterprise rules, intelligent agents, STEP, and workflow.” Architectures, Networks, and Intelligent Systems for Manufacturing Integration **3203**: 160-171.

Goldin, D., S. Venneri, et al. (1998). New Frontiers in Design Synthesis. 49th International Astronautical Congress, Melbourn, Australia, International Astronautical Federation.

Harry, M. J. and J. R. Lawson (1992). Six Sigma Producibility Analysis and Process Characterization. Reading, MA, Addison-Wesley Publishing Company, Inc.

Hazelrigg, G. A. An Axiomatic Framework for Engineering Design. Arlington, VA, National Science Foundation**:** 14.

Hazelrigg, G. A. (1998). “A Framework for Decision-based Engineering Design.” Journal of Mehanical Design **120**(December 1998): 653-658.

Hazelrigg, G. A. (1996). “The Implications of Arrow's Impossibility Theorm on Approaches to Optimal Engineering Design.” Journal of Mechanical Design **118**(June 1996): 161-164.

Kensinger, S. (1996). Re-Engineering Engineering: Promises-Promises. Computer-Aided Engineering**:** 53-58.

Kundu, A. K., J. K. Watterson, et al. (1998). A Multi-Disiplinary Study of Aircraft Aerodynamic Surface

Smoothness Requirements to Reduce Operating Cost. AIAA/USAF/NASA/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO.

Magrab, Edward B. Integrated Product and Process Design and Development. Boca Raton: CRC Press. 1997.

Malen, D. E. (1996). "Decision Making in Preliminary Product Design: Combining Economic and Quality Considerations." The Engineering Economist **41**(Winter 1996): 105-122.

Marston, M. and F. Mistree (1997). A Decision Based Foundation for Systems Design: A Conceptual Exposition.

McClure, R. W. (1999). "Design for manufacturability." IIE Solutions(November 1999): 41-44.

Megson, T. H. G. (1990). Aircraft Structures for Engineering Students. New York, Halsted Press.

Michaels, J. V. and W. P. Wood (1989). Design to Cost. New York, John Wiley & Sons.

Mistree, F., W. Smith, et al. (1990). "Decision-Based Design: A Contemporary Paradigm for Ship Design"." Transactions, Society of Naval Architects and Marine Engineers: 565-597.

Nicolai (1975). Fundamentals of Aircraft Design. San Jose, CA, METS inc.

O'Flynn, M. J. and M. M. Ahmad Computer-Based Concurrent Engineering Systems. Computer-Based Sysytems: A DFM Cost-Based System**:** 185-206.

Presley, A. and D. H. Liles (1995). The Use of IDEFO for the Design and Specification of Methodologies. 4th Industrial Engineering Reasearch Conference Proceedings, Nashville, TN, Institute of Industrial Engineers.

Pugh, Stuart. Total Design: Integrated Methods for Successful Product Engineering. Wokingham, England: Addison-Wesley Publishing Company. 1991.

Rais-Rohani, M., Ph.D., P.E. (1998). A Framework for Preliminary Design of Aircraft Structures Based on Process Information, Final Report - 2. Mississippi State, Mississippi State University**:** 33.

Rais-Rohani, P. D., P.E. (1998). A Framework for Preliminary Design of Aircraft Structures Based on Process Information, Final Report - 1. Mississippi State, Mississippi State University**:** 1-16.

Raymer, D. P. (1992). Aircraft Design: A Conceptual Approach. Washington, American Institute of Aeronautics and Astronautics.

Rouse, William B. Design for Success: A Human-Centered Approach to Designing Successful Products and Systems. New York: John Wiley & Sons, Inc. 1991.

Rummler, G. A. and A. P. Brache, Improving Performance, 2nd Edition, San Francisco: Jossey-Bass Publishers, 1995.

Unal, R. and E. B. Dean (1995). Design for Cost and Quality: The Robust Design Approach. **1999**.

Ulrich, Karl T. and Steven D. Eppinger. Product Design and Development. New York: McGraw-Hill, Inc. 1995.

Whitney, D. E. (1988). “Manufacturing by Design.” Harvard Business Review(July-August 1988): 83-91.

Yoshimura, M. (1993). Concurrent optimization of product design and manufacture. Concurrent Engineering: Contemporary issues and modern design tools. H. R. Parsaei. London, Chapman & Hall**:** 159-183.

Z., R.-R. M. a. H. (1999). Analysis and Optimization of Primary Aircraft Structures Based on Strength, Manufacturability, and Cost Requirements. 40th AIAA/ASME/ASCE/AHS/ASC Structures, Strutural Dynamics and Materials Conference, St. Louis, MO.

4.8.2.6 Design Tools

Mavris, D. N., O. Bandte, et al. (1999). “Robust Design Simulation: A Probabilistic Approach to Multidiciplinary Design.” Journal of Aircraft **36**(January- February 1999): 298-307.

Mayer, R. J. P. D. and P. S. P. D. deWitte Delivering Results:Evolving BPR from Art to Engineering, KBSI.

Parsaei, H. R. and W. G. Sullivan, Eds. (1993). Concurrent Engineering: contemporary issues and modern design tools. London, Chapman & Hall.

Rais-Rohani, M. and Z. Huo (1999). Analysis and optimization of primary aircraft structures based on strength, manufacturability, and cost requirements. 40th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics and Materials Conference, St. Louis, MO, American Institute of Aeronautics and Astronautics.

Walter, T. J. (1999). A Framework for Integrating Design Automation with Computer Aided Parametric Estimating (CAPE). Mount Laurel, NJ, Lockheed Martin PRICE Systems**:** 6.

4.8.2.7 Integrated Product and Process Development (IPPD)

DOD Guide to IPPD,
http://www.acq.osd.mil/te/survey/table_of_contents.html, 1996

Adler, P. S., A. Mandelbaum, et al. (1996). Getting the Most Out of Your Product Developement Process. Harvard Business Review**:** 4-15.

Augustine, N. R. (1981). “Augustine' Laws and Major System Development Programs.” : 64-91.

Belhe, U. and A. Kusiak The House of Quality in Analysis of Product Developement Process. 4th Annual Industrial Engineering Reasearh Confrence, Institute of Industrial Engineers.

Defense, D. o. (1985). Transition from Development to Production ... Solving the Risk Equation.

Washington, DC, Assistant Secretary of Defense - Acquisition and Logistics.

Garvin, D. A. (1984). “What Does "Product Quality" Really Mean?” Sloan Management Review(Fall 1984): 34-52.

Hale, M., J. Craig, et al. (1995). On the Development of a Computing Infrastructure that Facilitates IPPD from a Decision-Based Perspective. 1st AIAA Aircraft Engineering, Technology, and Operations Congress, Los Angeles, CA, American Institute of Aeronautics and Astronautics.

Hisakazu, S., K. Yasuhiko, et al. (1988). Using the Demanded Quality Function Deployment Chart: Developing the Quality Plan. Quality Function Deployment, Integrating Customer Requirements into Product Design. J. Akao, Productivity Press.

Jain, S. C. (1994). “Cost-Effective Process Modeling: Are We There Yet?” JOM: 16.

Keen, P. G. W. and E. M. Knapp (1996). Every Manager's Guide to Business Processes. Boston, Harvard Business School Press.

Marca, D. A. and C. L. McGowan (1993). IDEF0/SADT Business Process and Enterprise Modeling. San Diego, Eclectic Solutions.

Martin, J. N. (1997). Systems Engineering Guidebook: A Process for Developing Systems and Products. New York, CRC Press.

Mizuno, S. and Y. Akao, Eds. (1994). QFD: The Customer-Driven Approach to Quality Planning and Deployment. Tokyo, Asian Productivity Organization.

Nevins, J. L. and D. E. Whitney, Eds. (1989). Concurrent Design of Products and Processes. New York, Mc-Graw-Hill Book Company.

Rais-Rohani, M. and A. G. Greenwood (1998). Product and process coupling in multidisciplinary design of flight vehicle structures. 7th AIAA/NASA/USAF/ISSMO Symposium on Multidisciplinary Analysis and Optimization, St. Louis, MO, American Institute of Aeronautics and Astronautics.

Schrage, D. P. and D. N. Mavris (1995). Integrated Product/Process Design/Development (IPPD) Through Robust Design Simulation: The Key for Affordable Systems. 1st AIAA Aircraft Engineer, Technology, and Operations Congress, Los Angeles, CA, American Institute of Aeronautics and Astronautics.

Weber, N. O. (1994). Product Developement Teams and Tools Applied to the Aircraft Industry. World Class Design to Manufature. **1:** 20-26.

4.8.2.8 Knowledge Management

(1999). Do We Know How to Do That? Understanding Knowledge Management. Harvard Management Update**:** 3-5.

Coyne, R. D., M. A. Rosenman, et al. (1990). Knowledge-Based Design Systems. Reading, MA, Addison-Wesley Publishing Company.

Hess, T. J. (2000). “Using Autonomous Software Agents to Create the Next Generation of Decision Support Systems.” Decision Sciences **31**(1): 1-31.

Jacobs, P. (1999). Data Mining: What General Managers Need to Know. Harvard Management Update**:** 4.

Liebowitz, J. and L. C. Cox (1997). Knowledge Management and Its Integrative Elements. Boca Raton, CRC Press.

Su, S. Y. W. (1995). An Extensible Knowledge Base Management System for Supporting Rule-based Interoperability among Heterogeneous Systems. Database Systems Research and Development Center. Gainesville, University of Florida**:** 10.

Zack, M. H. (1999). “Managing Codified Knowledge.” Sloan Management Review(Summer 1999): 45-58.

4.8.2.9 Manufacturing Processes

Machinery's Handbook 25: The Bible of the Mechanial Industries, Industrial Press. 1998.

Benedict, G. F. (1987). Nontraditional Manufacturing Processes. New York, Marcel Dekker, Inc.

Center, A. S. (1993). Manufacturing Development Guide. Wright-Patterson Air Force Base, OH, Aeronautical Systems Center, Air Force Materiel Command.

Creese, R. C., M. Adithan, et al. Estimating and Costing for the Metal Manufacturing Industries. New York, Marcel Dekker, Inc.

Foley, M. and E. Bernardon (1991). Thermoplastic Composite Manufacturing Cost Analysis. International Encyclopedia of Composites. S. M. Lee. New York, VCH Publishers. **6:** 1-10.

Fondon, J. W., Jr., D. Rasmussen, et al. (1996). Large Composite Structure -- Commercial Military Integration (Military Products Using Best Commercial Practices) Phase I -- Manufacturing 2005 Business Practices, Manufacturing Infrastructure, Process Technology Development, Volume II -- Manufacturing 2005 Handbook. Wright-Patterson Air Force Base, OH, Manufacturing Technology Directorate, Wright Laboratory, Air Force Materiel Command.

Gutowski, T., R. Henderson, et al. (1991). “Manufacturing Costs for Advanced Composites Aerospace Parts.” SAMPE Journal **27**(3 (May/June 1991)): 37-43.

Hsu, C. and L. Rattner (1990). “Information Modeling for Computerized Manufacturing.” IEEE Transactions on Systems, Man, and Cybernetics, **20, #4**(July/August 1990): 758-776.

Kalpakjian, S. (1992). Manufacturing Engineering and Technology. Reading, MA, Addison-Wesley Publishing Company.

Kessler, W. C., G. Shumaker, C., et al. (1993). Early Manufacturing Cosiderations in Design. AGARD Meeting in 'Integrated Airframe Design Technology'.

Kusiak, A., T. N. Larson, et al. (1994). "Re-engineering of Design and Manufaturing Process." Computers and Industrial Engineering **26**(3): 521-536.

Kutay, A. and S. Finger (1991). Integration of economic information with design and manufacturing systems. Pittsburgh, Pa., Carnegie Mellon University The Robotics Institute.

Malstrom, E. M., Ed. (1984). Manufacturing Cost Engineering Handbook. New York, Marcel Dekker, Inc.

Moore, H. D. and D. R. Kibbey (1982). Manufacuring: Materials and Processes. New York, John Wiley & Sons.

Nau, D. S., G. Zhang, et al. (1993). Evaluating product machinability for concurrent engineering. Concurrent Engineering: Contemporary issues and modern design tools. H. R. Paraei and W. G. Sullivan. London, Chapman & Hall**:** 264-279.

Noton, B. R. (1988). Manufacturing Cost/Design Guide for Aerospace Applications. Columbus, OH, Battellle Columbus Division.

Noton, B. R. (1989). Manufacturing Cost/Design Guide for Conceptual Design. Columbus, OH, Battellle Memorial Institute.

Rais-Rohani, M. and E. B. Dean (1996). Toward Manufacturing and Cost Consideration in Multidiciplinary Aircraft Design. The 37th AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference, Salt Lake City, UT, American Institute of Aeronautics and Astronautics.

Rais-Rohani, P. D., P.E. Recent Progress in Modeling Manufaturability Factors for Multidiciplinary Design of Aircraft Structures. Mississippi State, MS, Mississippi State University**:** 1-22.

Riseley, M. K., T. Houpt, et al. (1996). Large Composite Structure -- Commercial Military Integration (Military Products Using Best Commercial Practices) Phase I -- Manufacturing 2005 Business Practices, Manufacturing Infrastructure, Process Technology Development, Volume I -- Program Review. Wright-Patterson Air Force Base, OH, Manufacturing Technology Directorate, Wright Laboratory, Air Force Materiel Command.

Sehdev, K., I.-S. Fan, et al. (1995). Design for Manufacture in the Aerospace Extended Enterprise. World Class Design to Manufacture. **2:** 28-33.

Shah, J. J., M. Mantyla, et al. (1994). Advances in Feature Based Manufacturing. Amsterdam, Elsevier.

Silan, M. N., M. Stanton, et al. (1993). Manufacturing System Modeling and Analysis Applied Within a Theory of Constaints Framework. Norfolk, VA, Old Dominion University**:** 169-175.

Simpson, J., L. Field, et al. (1991). The Boeing 767: From Concept to Production (A). Boston, Harvard College**:** 19.

Sims, E. R., Jr. (1995). Precision Manufacturing Costing. New York, Marcel Dekker, Inc.

Stoll, H. W. (1986). "Design for Manufacture: An Overview." Applied Mechanics **39**(9): 1356-1363.

Strong, A. B. (1989). Fundamentals of Composite Manufacturing: Materials, Methods, and Applications. Dearborn, MI, Society of Manufacturing Engineers.

Todd, R. H. (1994). Manufacturing Process Reference Guide. New York, Industrial Press Inc.

Yankee, H. W. (1971). Manufacturing Processes. Englewood Cliffs, NJ, Prentice-Hall, Inc.

Shankar, S. R. and D. G. Jansson A generalized methodology for evaluating manufacturability.

4.8.2.10Materials

Ashby, M. F. Materials Selection in Mechanical Design. New York, Pergamon Press.

Ermanni, P. and G. Ziegmann (1991). Cost-Efficiency of Highly Integrated Fuselage Structures: Comparison Between Metals and Composites. 12th International European Chapter Conference of the Society for the Advancement of Material and Process Engineering, Maastricth, The Neatherlands, SAMPE/Elsevier Science Publishers B.V.

Farag, M. M. (1979). Materials and Process Selection in Engineering. London, Applied Science Publishers Ltd.

Hoskin, B. C. and A. A. Baker (1986). Composite Materials for AIrcraft Structures. New York, American Institute of Aernautics and Astronautics.

Resetar, S. A., J. C. Rogers, et al. (1991). Advanced Airframe Structural Materials: A Primer and Cost Estimating Methodology. Santa Monica, CA, RAND Corp.

4.2.8.11Object-Oriented Technology

Object Management Group, http://www.omg.org, 1998,1999

Tutorial on Distributed Objects, http://www.expertsoft.com/Resources/DistTech/tutorial.html, 1998

Anemaat, W., A. and K. L. Schueler (1997). Airplane Configuration Layout Design Using Object-Oriented Methods. World Aviation Congress, Anaheim, CA, American Institute of Aeronautics and Astronautics.

Greenwood, A. G. and S. R. Guo (1999). “Integrating Cost Assessment into Product/Process Design: An Object-Based Approach.” Journal of Engineering Valuation and Cost Analysis (Sept 1999): 8.

Greenwood, A. G. (1997). An Object-Based Approach for Integrating Cost Assessment into Product/Process Design. Bolling Air Force Base, DC, Air Force Office of Scientific Research.

Greenwood, A. G. (1999). Integrating Cost Assessment Into Product/Process Design Through an Object-Based Approach. Decision Sciences Institute Conference, New Orleans.

Hathaway, Bob, object-orientation FAQ, http://www.cyberdyne-object-sys.com/oofaq/, 1996

Jackson, R. B., W. C. Giauque, et al. (1996). "Concepts of Single-Paradigm Object-Oriented Developement, with Application to a Manufacturing Information System." IEEE Transactions on Systems, Man, and Cybernetics **26**(Sept, 96): 583-593.

Jacobson, I., M. Ericsson, et al. (1995). The Object Advantage: Business Process Reengineering with Object Technology. Reading, MA, Addison-Wesley Publsihing Company.

Kemper, A. and G. Moerkotte (1994). Object-Oriented Database Management: Applications in Engineering and Computer Science.

Khoshafian, S. and R. Abnous (1990). Object Orientation Concepts, Languages, Databases, User Interfaces. New York, John Wiley & Sons, Inc.

Kulak, D. a. G., Eamonn (2000). Use Cases, Requirements in Context. New York, New York, ACM Press.

Lewis, G. R., *CORBA 2.0: Universal Networked Objects*, Standard View, vol. 3, no. 3, 102-106, 1995

Marshall, C. (2000). Enterprise Modeling with UML, Addison Wesley Longman.

Martin, J. and J. J. Odell Object-Oriented Analysis and Design. Englewod Cliffs, NJ, Prentice Hall.

Mize, J. E., H. C. Pratt, et al. (1992). "Modeling of Integrated Manufacturing Systems Using an Object-Oriented Approach." IIE Transactions **24**(July, 92): 14-26.

Monk, S., J. A. Mariani, et al. (1996). Migration from Relational to Object-Oriented Databases. Information and Software Technology, Elsevier Science B. V.**:** 467-475.

Orfali, R., D. Harkey, and J. Edwards, *CORBA, Java and the Object Web*, BYTE Magazine, October 1997.

Shlaer, S. and S. J. Mellor (1988). Object-Oriented Systems Analysis: Modeling the World in Data. Englewood Cliffs, NJ, Yourdon Press.

Wu, B. (1995). "Object-oriented systems analysis and definition of manufacturing processes." International Journal of Production Research **33**(4): 955-974.

Yang, Z. and K. Duddy CORBA: A Platform for Distiributed Object Computing. CRC for Distributed Systems Technology, University of Queensland**:** 28.

4.8.2.12Information Technology

CORBA Overview, http://www.infosys.tuwien.ac.at/Research/Corba/OMG/arch2.html, 1998

Interactive Connections: Internet Glossary, http://www.icactive.com/_internetglossary.html

NIIIP (National Industrial Infomration Infrastructure Protocols), http://www.niiip.org, 1998,1999

STEP - ISO 10303, http://www.ukcic.org/step/step.htm, 1997

Tech-Web Encyclopedia, http://www.techweb.com/encyclopedia/, 1999

The ISO STEP Standards, http://www.steptools.com/library/standard/, 1997

Aamodt, A. and E. Plaza, Case-Based Reasoning: Foundational Issues, Methodological Variations, and System Approaches, AI Communications, IOS Press, vol.7, no.1, 39-59, 1994

Allen, B. P., Case-Based Reasoning: Business Applications, Communications of the ACM, vol.37, no.3, 40-42, 1994

Balen, H., CORBA and the WWW, Addendum to the 1997 ACM SIGPLAN conference on Object-oriented programming, systems, languages, and applications (Addendum) , 128-132, 1997

Emmerich W., An Architecture for Viewpoint Environments based on OMG/CORBA, Joint proceedings of the second international software architecture workshop (ISAW-2) and international workshop on multiple perspectives in software development (Viewpoints '96) on SIGSOFT '96 workshops , 207-211, 1996

Emmerich W., *An Introduction to OMG/CORBA*, Proceedings of the 1997 international conference on Software Engineering , 641-642, 1997

Erkes, J. W., K.B. Kenny, J. W. Lewis, B. D. Sarachan, M. W. Sobolewski, and R. N. Sum, *Implementing Shared Manufacturing Services on the World-Wide Web,* Communications of the ACM, vol. 39, No. 2, 34-45, 1996

Goldschmidt, A., Report on NIIIP, Communications of the ACM, vol. 39, no. 3, p100-103, 1996

Hardwick, M., D. L. Spooner, et al. (1997). "Data Protocols for the Industrial Virtual Enterprise." IEEE Journal of Internet Computing(Jan/Feb 1997): 1-12.

Hardwick, M., D. L. Spooner, T. Rando, and K.C. Morris, Sharing Manufacturing Information in Virtual Enterprises, Communications of the ACM, vol. 39, no. 2, 46-54, 1996

Hardwick, M., R. Bolton, The Industrial Virtual Enterprise, Communications of the ACM, vol. 40, no. 9, 59-60, 1997

Perera, S. and I. Watson, Nirmani: An Integrated Case-Based System for Strategic Design and

Estimating, Lecture Notes in Artificial Intelligence (1020) Ed. by Watson, I., 185-200, Berlin: Springer,

1995

Prasad, M. V. N., V. R. Lesser, and S. E. Lander, *Retrieval and Reasoning in Distributed Case Bases*, Hournal of Visual Communication and Image Representation, Special Issue on Digital Libraries, vol7, no.1, 74-87, 1996

Schofield, C. S., and M. Shepperd, *Software Support for Cost Estimation by Analogy*, ESCOM6, Rolduc, 1995

Slade, S. (1991). Case-based Reasoning: A Research Paradigm. A I Magazine. **12:** 42-55.

Vernadat, F. B. (1996). Enterprise Modeling and Integration: Principles and Applications. London, Chapman & Hall.

Watson, I., *Applying Case-Based Reasoning: Techniques for Enterprise Systems*, San Francisco: Morgan Kaufmann Publishers, Inc., 1997

Fielding, R. T., E. J. Whitehead, Jr., et al. (1998). “Web-based Development of Complex Information Products.” Communications of the ACM **41**(August, '98): 84-92.

4.8.2.13Reliability Prediction

Denson, W. (1998). “The History of Reliability Prediction.” IEEE Transactions Reliability **47**: 321-328.

Fajdiga, M. (1996). “Reliability Prediction in Early Phases of Product Design.” Journal of Engineering Design **7**: 107-129.

Morris S.F., R. J. F. (1993). MIL-HDBK-217 - A Favorite Target. Proc. Ann. Reliability and Maintainability Symp.

Relex Software Corporation, “Reliability Prediction”, www.relexsoftware.com/predict.htm.

Spencer,J. L. “The High and Lows of Reliability Predictions”, Proceedings of the Annual Reliability and Maintainability Symposium, 1986, pp 156 – 159.

P.T. O’Connor, P. T., M.G. Head, and M. Joy, “Reliability Prediction for Microelectronic Systems”, Proceedings of the Annual Reliability and Maintainability Symposium, 1983, pp 452 – 454.

4.8.2.14Requirements Management

Jones, D. A. H., Frank - Kar, Pradip C. - Bell, Marty (1997). Interfacing Requirements Management Tools In The Requirements Management Process - A First Look. Proceedings of the Seventh International Symposium of the INCOSE.

4.8.3 Appendix 4-C: IDL Specification for Prototype

IDL (Interface Definition Language) is a specification language, not a implementation language. Its purpose is to define the capabilities of a distributed service along with a common set of data types for interacting with those distributed services. IDL meets a number of objectives:

- Language independence
- Distributed service specification
- Definition of complex data types
- Hardware independence

IDL provides flexibility for programmers. It allows programmers to choose the most appropriate operating system, execution environment and even programming language to use for each component of a system under construction. More importantly, IDLs allow the integration of existing components, such as existing cost estimation systems.

```
//Constants-Typedefs-exceptions

module ICEM{

interface ICEM_Item;
interface ICEM_ItemContainer;
interface ICEM_CostItem;
interface ICEM_CostCategory;

 enum ServiceType{
 COST_ESTIMATION_, CADDRAWING_, MANUFACTURABILITY_
 };

 enum ModelType{
 PARAMETRIC, NON_PARAMETRIC
 };

 enum FeatureInputType{
  EDIT,      //Single line edit box
  PDM  //Pull Down Selection box
  };

 enum FeatureValueType{
    INT_,           DOUBLE_,     STRING_
   };

 enum ICEM_ExceptionType{
             SERVER_COMM_ERROR,
    WRONG_ITEMTYPE
             };
 typedef sequence<string> StringCollection;

  struct InputType{
   FeatureInputType type;  //To identify input type
   StringCollection options;     //Options for user to select as input.
   StringCollection options_value;
   };

  struct Feature{
   string Description;     //Description of feature
   string Value;           //Value of this feature
   string Unit;            //Unit of this feature
   InputType Input;        //object to direct how to input this features. Defined
later.
   FeatureValueType Value_Type;         //To identify value type
   boolean Read_Only;
  };

 struct Model{
   string Name;                //Name of this model
   string Description;     //Other info that might be interested to Designer
   string Version;         //Version information
   string ModelContext;    //Unique Model Context in CORBA environment
   long Year;              //Identify the calculation based on which year:
   ModelType Type;  //Type of Model: parametric/non-parametric, see constants
  };

  struct CostValue{
   double Labor;           //Labor Cost
   double Material;        //Material Cost
   double Capital;  //Capital Cost
   double Total;           //Total Cost : may not be equal to M+L+C
  };

       //TypeDefs
       typedef sequence<ICEM_Item> ItemCollection;
       typedef sequence<Feature> FeatureCollection;
       typedef sequence<ICEM_CostItem> CostItemCollection;
       typedef sequence<Model> ICEM_Model_Collection;

       //Exceptions
       exception ICEMException{
             ICEM_ExceptionType error;
             string description;
  };
```

//Basic Data Representations:

```
interface ICEM_ItemProperty{
 enum IPType{
 MATERIAL, FORM, PROCESS, FEATURES
 };
 attribute FeatureCollection fs;               //Feature is an object defined later
 attribute IPType type;                 //To identity ItemProperty Type:
Material/Form/Process
// See Figure 1 for constants for this attribute.
};

interface ICEM_Material :ICEM_ItemProperty{          //Inherit ItemProperty , First
Feature is the name
};

interface ICEM_Form :ICEM_ItemProperty{        //Inherit ItemProperty, First Feature
is the name
};

interface ICEM_Process :ICEM_ItemProperty{           //Inherit ItemProperty, First
Feature is the name
};

interface ICEM_Features:ICEM_ItemProperty{           //Inherit ItemProperty
};
```

```
interface ICEM_Item{
  enum ICEM_ItemType {
  UNIT, ASSMEMBLY, CONTAINTER
  };
 attribute string Name;
 attribute long BaseQuantity;           //Expected Design quantity
 attribute ICEM_ItemContainer Parent; //Null represents that it is a root item.
 attribute ICEM_CostCategory c;  // item’s cost.
//Cost is an object defined later.
 attribute ICEM_ItemType type;   //To identify ItemType: Unit Item, Assembly or Item
Container
// See Figure 1 for constants for this attribute.
 long getTotalQuantity();
};

interface ICEM_UnitItem :ICEM_Item{            //Inherit Item
 attribute ICEM_Material m;
 attribute ICEM_Form f;
 attribute ICEM_Process p;
 attribute ICEM_Features fs;     //Additional Features required by cost estimation
 attribute Model modelinfo;      //Model Information, Defined later.
};

interface ICEM_ItemContainer :ICEM_Item{      //Inherit Item
 attribute ItemCollection items;        //Items within this container.
};

interface ICEM_Assembly :ICEM_Item{     //Inherit Item
 attribute ItemCollection Items;        //Items to be assembled
 attribute ICEM_Features fs;
 attribute Model modelinfo;      //Model Information, Defined later.
};

interface ICEM_Design :ICEM_ItemContainer{           //Inherit ItemContainer
 attribute long year;
 attribute string Memo;
};

interface ICEM_CostItem{
 enum ICEM_CostItemType{
      BASIC_ELEMENT, CATEGORY
 };

 enum CostValueType{
 RECURRING, NON_RECURRING, DEPENDS
 };

 attribute string Name;
 attribute CostValue cost;
 attribute ICEM_CostItemType type;
 attribute CostValueType value_type;
 attribute boolean required;
};

interface ICEM_BaseCostElement :ICEM_CostItem{
};
interface ICEM_CostCategory :ICEM_CostItem{
 attribute CostItemCollection CostElements;
};

//Cost Estimation Model Services
```

```
interface ICEM_ModelAgent{

 ICEM_Model_Collection ModelAvailable(in ICEM_Item item); //raises (ICEMException);
// ICEM_Model_Collection ModelAvailable(in ICEM_UnitItem item); //raises
(ICEMException);
//ICEM_Model_Collection ModelAvailable(in ICEM_Assembly item); //raises
(ICEMException);
//Check for the availability of the Model.
//Only accept UnitItem and Assembly as Input
//Model.Year = -1 indicates Model unavailable
 void getModelInterface(inout ICEM_Item item) ;//raises (ICEMException);
// void getModelInterface(inout ICEM_UnitItem item) ;//raises (ICEMException);
//void getModelInterface(inout Assembly item) ;//raises (ICEMException);
//Fill in the item features field
//Only accept UnitItem and Assembly as Input
 void CostEvaluation(inout ICEM_Item item);// raises (ICEMException);
// void CostEvaluation(inout ICEM_UnitItem item);// raises (ICEMException);
//void CostEvaluation(inout ICEM_Assembly item);// raises (ICEMException);
 //Only accept UnitItem and Assembly as Input
 };
```

//Data Dictionary Services:

```
interface ICEM_DataDictionary{
 enum DataDictionaryCategory{
      MATERIAL, FORM, PROCESS, ASSEMBLY, COST_COMPONENT
};
 boolean IsInDataDictionary(in string Word,
      in DataDictionaryCategory category);
//raises (ICEMException);
 boolean IsSubClassificationof(in string Word,
          in string Classification,
          in DataDictionaryCategory category);
//raises (ICEMException);
};
```

//CAD Drawing Service

```
interface ICEM_CADDrawing{
 boolean IsDrawingAvailable(in ICEM_Item item);// raises (ICEMException);
 void getDrawingInterface(inout ICEM_Item item);// raises (ICEMException);
 string getCADDrawingURL(in ICEM_Item item);// raises (ICEMException);
//Return URL of the final drawing
};
```

//Manufacturability Evaluation Service

```
interface ICEM_Manufacturability{
//Not yet Defined
};
};
```

[i] Fabrycky, W.J., Blanchard, B.S., *Life-Cycle Cost and Economic Analysis*, Prentice-Hall, Inc., 1991

[ii] ACEIT documentation

[iii] MAAP documentation

[iv] Stewart, R.D., Wyskida, R.M., Johannes, J.D., *Cost Estimator's Reference Manual,* John Wiley & Sons, Inc., 1995